DOCUMENT RESUME

ED 040 884 24 SE 009 215

AUTHOR	Dasenbrock, David; Teates, Thomas
TITLE	CAI Programs and Macros for Formative Curriculum Evaluation.
INSTITUTION	Florida State Univ., Tallahassee. Dept. of Science Education.
SPONS AGENCY	National Science Foundation, Washington, D.C.; Office of Education (DHEW), Washington, D.C. Bureau of Research.
REPORT NO	TM-1
BUREAU NO	BR-6-1762
PUB DATE	1 Jun 70
CONTRACT	OEC-2-6-061762-1745
NOTE	76p.
AVAILABLE FROM	Intermediate Science Curriculum Study, Dept. of Science Education, Florida State University, Tallahassee, Florida
EDRS PRICE	EDRS Price MF-$0.50 HC-$3.90
DESCRIPTORS	*Computer Assisted Instruction, Computer Programs, *Curriculum Development, *Curriculum Evaluation, Evaluation, Programing, *Secondary School Science

ABSTRACT

This technical memo is designed for persons who are interested in research with and development of curriculum material. Detailed information regarding computer programs, and program documentation used in the development and evaluation of ISCS curriculum materials is provided. The memo includes supplemental information to ISCS Technical Report I entitled "CAI Utilization for Formative Curriculum Evaluation." Flow diagrams and program listings of the ten types of CAI programing macros as well as the five data analysis programs developed by ISCS are included. This material also provides analysis programs for the formative evaluation of CAI text. (Author/FL)

TECHNICAL MEMO I
CAI PROGRAMS AND MACROS FOR FORMATIVE CURRICULUM EVALUATION

Prepared By

David Dasenbrock
and
Thomas Teates

June 1
1970

The research reported herein was performed under USOE Contract OE 2-6-061762-1745 from June 1966 to March 1970 and supported by The National Science Foundation Grant GW 4235.

INTERMEDIATE SCIENCE CURRICULUM STUDY
DEPARTMENT OF SCIENCE EDUCATION
FLORIDA STATE UNIVERSITY
TALLAHASSEE, FLORIDA

ACKNOWLEDGMENTS

Many of the macros and analysis programs described in this memo were designed by the authors. Mr. Paul Flood assisted with the design of several macros and contributed valuable suggestions and constructive criticisms during the development of the analysis programs.

At the FSU-CAI Center, Dr. Duncan Hansen, Center Director, and Dr. Walter Dick provided encouragement and advice as well as making their facilities available for the development and use of these programs. Much credit is due Mrs. Betty Wright for her frequent and most helpful suggestions regarding macro designs and programming details. Mr. George Hogshead provided valuable assistance with the development of analysis programs.

All inquiries regarding this memo should be addressed to:

Dr. David D. Redfield, Associate Director
Intermediate Science Curriculum Study
Department of Science Education
The Florida State University
Tallahassee, Florida 32304

FORWARD

The ISCS Tech Memo series, like the ISCS Technical Report series, is intended to provide communication to other colleagues and interested professionals who are actively interested in research with and development of curriculum material.

The rationale for the Technical Report series is three fold. First, to report in a concise, descriptive, and explanatory nature advances made in the technology of curriculum development. Secondly, pilot studies that show great promise with potential for further research and subsequent reporting can be given quick distribution. Third, the Technical Report series provides for distribution of pre-publication copies of implementation studies that, after proper technical review, will ultimately be found in professional journals.

The Tech Memo, on the other hand, serves a supporting documentation function by providing detailed information such as computer programs, program documentation, etc. with sufficient explanatory sections to enable the use or adaptation of the material presented.

This Tech Memo provides supplemental information to ISCS Technical Report 1 entitled "CAI UTILIZATION FOR FORMATIVE CURRICULUM EVALUATION." Included herein are Flow Diagrams and program listings of the ten types of CAI programming macros as well as the five data analysis programs developed by ISCS which were described in the report. This material is offered in the spirit of making available information which should significantly reduce the effort required to program curriculum material. In addition, it provides potentially useful analysis programs for the formative evaluation of CAI text.

Ernest Burkman, Director
Intermediate Science Curriculum Study

June 1, 1970
The Florida State University
Tallahassee, Florida

GENERAL BACKGROUND ON THE INTERMEDIATE SCIENCE CURRICULUM STUDY

The Intermediate Science Curriculum Study (ISCS) is a large-scale instructional research project supported to date by a contract with the United States Office of Education and grants from the National Science Foundation. The project is designed to develop, test, and disseminate into practice a system of individualized science instruction for grades seven through nine.

The project is organized on a develop-field-test-revise design. Draft materials are produced at Florida State University by on- and invited off-campus personnel and tested on a large national sample of junior-high-school students. During the 1969-70 school year, more than 75,000 students in 25 states are involved in the field testing of the ISCS materials. In addition, a small number of students from the Florida State University campus school are taking a computer-assisted instruction version of the materials from which additional feedback data are being accumulated. To date, more than 400 scientists, teachers, and education specialists have cooperated in the development process.

The most unique feature of the ISCS materials is the fact that the students using them progress at different rates and through different instructional pathways depending upon their interests, abilities, and previous experience. The materials are being designed that this can be accomplished in ordinary science classrooms by teachers with limited special training.

The package of instructional materials for each grade level consists of student printed materials, specially designed laboratory apparatus, a student self-evaluation system based upon behavioral objectives established for the instructional materials, teacher orientation materials, and standardized tests. The Silver Burdett Corporation, in conjunction with Damon Educational Corporation, is distributing these materials during the experimental phase of the project and will market the commercial version of them.

The project has generated world-wide interest and its newsletter, published twice yearly, now goes to more than 10,000 people in 42 countries. ISCS materials are now in use in Australia and will be used in American dependent schools in Germany and Japan in September. Experimental testing of the materials is now underway in Manila, and plans have been established for a joint Florida State University - Philippines effort to produce a special Philippines version of the program. In addition, project personnel have visited Japan, India, and several South American countries for preliminary discussions related to possible use of the materials in these areas.

TABLE OF CONTENTS

LIST OF FIGURES

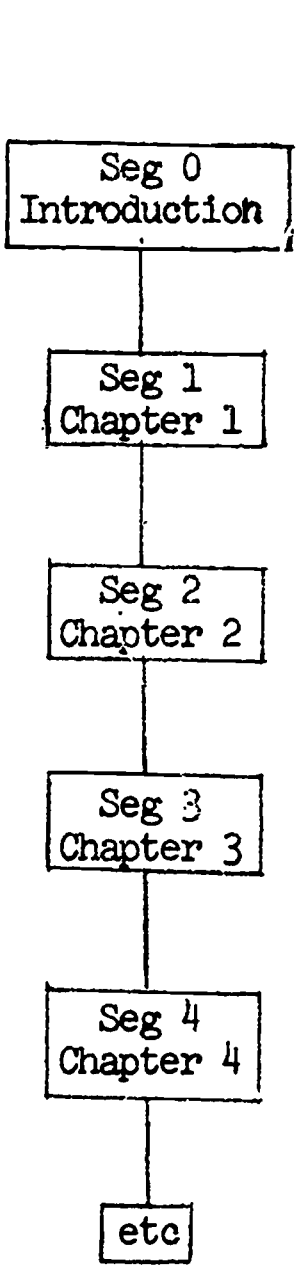
INTRODUCTION

For four years the Intermediate Science Curriculum Study has utilized computer assisted instruction as a relatively new technique for curriculum evaluation. The purpose of the CAI trial was to obtain objective data to aid in the revision of the ISCS texts used in the regular classroom. To attain this goal, a computer assisted instruction course was developed, that paralleled as closely as possible the ISCS individualized course materials in use in the regular classroom. The production of the CAI version of the ISCS instructional materials was a by-product of the basic research effort of curriculum evaluation and revision. A general description of the development and utilization of CAI by ISCS is presented in Technical Report 1 "CAI Utilization for Formative Curriculum Evaluation" available from ISCS.

This memo presents the technical details of the macros and analysis programs by which ISCS implemented and analyzed the most recent (1968-69) CAI versions of the entire level I (grade seven) and level II (grade eight) materials and one-half of the 1969-70 level III (grade nine) materials. In addition, details of content, processes, and other frame identifiers codes useful for analysis are described.

The computer programs are basically linear. The courses consist of a basic core, which all students follow. Excursions are provided, which are branches away from the core sequence, and the core is re-entered from where it was left. The students are able to go forward or backward in the materials, and are not limited to going forward only. The students therefore are able to go back and change answers at any time. The program is non-reinforcing, and is completely macro based. The macro system presents the student's last answer when the previous frame is reviewed.

The general "flow" of student progress is linear through either of the three levels of ISCS materials. Figure 1 illustrates the basic design of this linear flow from segment to segment of the program. Usually, each segment within a level represents one chapter of text material and the excursions designed to accompany that chapter. In some instances it was possible for students to review to an excursion in an earlier segment when remedial work was desired. (For example, a branch to an excursion on graphing (1-6) was available from either segment 3 or segment 10 of the level 1 program).



SEGMENT FLOW DIAGRAM
FIGURE 1

The Macros

General Descriptions

It has been said that the ISCS programs are "completely macro based." It is probable that the extensive evaluation completed for each grade level of the curriculum could not have been completed without the utilization of this very useful capability of the Coursewriter II language and the IBM 1500 system.

The following is a brief description of each macro. A more comprehensive description of the macros and their functions is given in ISCS Technical Report 1 -- "CAI Utilization for Formative Curriculum Evaluation".

Macro	Mode*	Type of Text Programmed	Branches
t&d500	LP	Information Presentation	Continue Review
exd500	LP	Branch to Excursion decision Frame	Excursion Next frame Review
ysd500	LP	Question whcih has a yes or no answer. Yes is the correct response. (A Variation provides for no as a correct response.)	Next Frame (with selection of answer) Review
mad500	LP	Multiple choice questions, choice A is correct answer. (Variations provide for B, C, or D as correct answer.)	Next frame (via an answer) Review
rst500	LP	Resource decision frame. (Request one of 32 resources, used in geology unit only.)	Continue (to next frame) Call for resource Review
tst500	LP	Test question. (Any one of 4 possible answers can be programmed as "correct.")	Skip (to next question) Review (to previous question) Respond (to one of four possible answers).

dcd500	KB	Free response questions. No specified correct answer. (Limited to a one line response.)	Continue (by entering a response) Review
cmt500	KB	Free Response--no specified correct answer. May be from one to ten lines in length.	Continue (by entering a response) Review
lmd500	KB	Free response question with numerical answer. Limit function used to determine if an answer is within certain limits and therefore acceptable	Continue (by entering an answer) Review
kld500	KB	Free response question with correct alphanumeric answer. Keyletter function used to determine whether a response is acceptable.	Continue (by entering an answer) Review

*LP = light pen; KB = keyboard

Assignments for Counters, Switches and Return Registers

The use of counters, switches, and return registers was standardized as much as possible throughout the macros in order to facilitate and expedite programming. The usual assignments for these record keeping devices are given below.

Counter Assignments

c1	correct answers
c2	incorrect answers
c3	total reviews
c7	current number of reviews
c20-30	control of display of previous answer upon review

Switch Usage

s20-s30	control of display of previous answer upon review
s10	control of branching in treatment and control groups for 1969-70 level III materials

Return Register Usage

rr2	return from excursions
rr4	return from graphic data display and resource display

Detailed Listings and Flow Diagrams

For each macro described above, this section sets forth the macro parameters, the Coursewriter II statements used to construct the macro, and a flow diagram for the macro.

ttd500
parameters
$01 review frame label
$02 EPID

TTD macro expanded

```
MACWTR
        MA   TTD500 $     00000
        DTI  30,0 / /6,0 /$01
        LR   $01 /RR4
        DTI  27,26 / /8,26 / / REVIEW
        DTI  27,3 / /19,3 / / READY TO CONTINUE
        PA   30
        EPP  9999 /$02
        NX
        BR   RE
        CAP  4,26,3,2 /TM
        BR   PR1
        AAP  4,26,3,25 /RV
        BR   $01
        UN   UN
        BR   RE
        EM
```

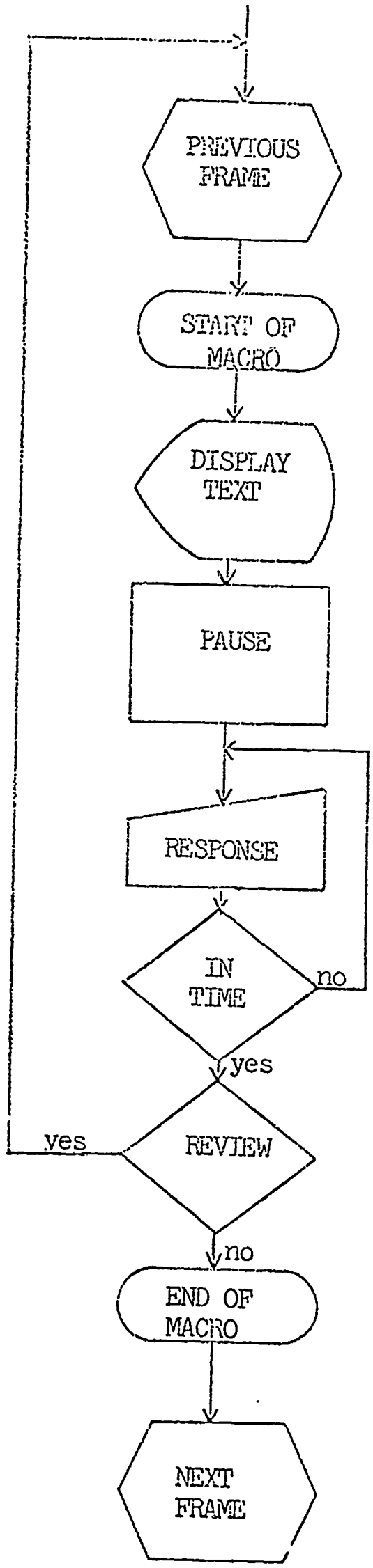


FIGURE 2
FLOW DIAGRAM FOR MACRO ttd500

exd500
parameters

$01	review frame label
$02	next frame label
$03	excursion frame label
$04	EPID

EXD macro expanded

```
MACWTR
        MA   EXD500 $     00000
        DT   28 / / /   /  R EVIEW
        DT   20 / / /   /  I  WILL DO THE  E XCURSION
        DT   24 / / /   /  I  WILL NOT DO THE  E XCURSION
        PA   30
        EPP  9999 /$04
        NX
        BR   RE
        CAP  4,19,3,0 /EX
        LR   $02 /RR2
        BR   $03
        CAP  4,23,3,0 /TM
        BR   $02
        AAP  4,27,3,0 /RV
        BR   $01
        UN   UN
        BR   RE
        EM
```

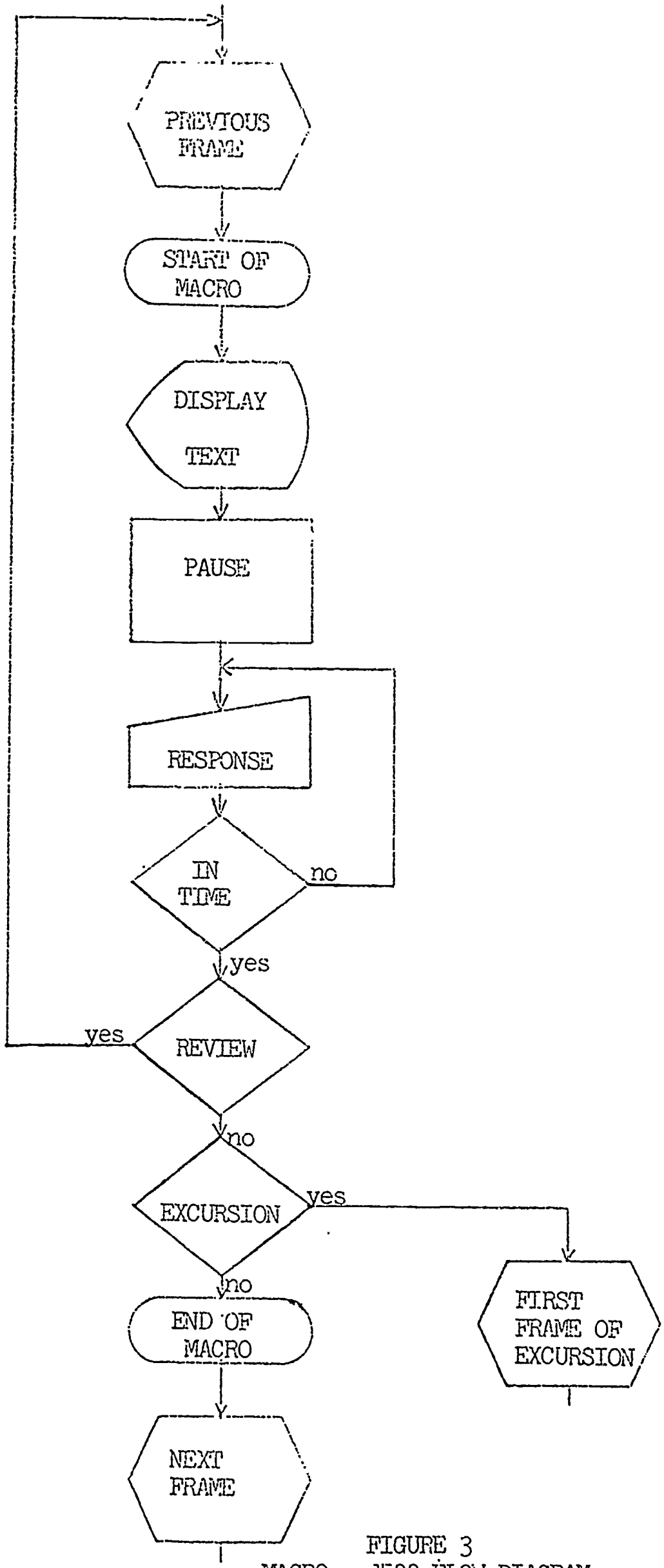


FIGURE 3
MACRO exd500 FLOW DIAGRAM

ysd500
parameters

$01	review frame label
$02	switch and counter number for this frame
$03	switch and counter number for the next frame
$04	EPID

YSD macro expanded

```
        MA   YSD500 $     00000
        DTI  27,2 / /5,2 / / YES
        LR   $01 /RR4
        DTI  27,16 / /4,16 / / NO
        DTI  27,31 / /8,31 / / REVIEW
        BR   #11 /C7 /G /9
        BR   #11 /S$02 /0
        BR   #12 /C$02 /E /2
        DTI  27,14 / /1,14 /X
        BR   #11
#12
        DTI  27,0 / /1,0 /X
#11
        PA   30
        EPP  9999 /$04
        NX
        BR   RE
        CAP  4,26,3,1 /CA
        LD   2 /C$02
        AD   1 /C1
        BR   #16
        AAP  4,26,3,30 /RV
        AD   1 /C7
        AD   1 /C3
        BR   $01
        WAP  4,26,3,15 /WN
        LD   1 /C$02
        AD   1 /C2
        BR   #16
        UN   UN
        BR   RE
#16
        BR   #18 /C7 /G /0
        LD   1 /S$02
        LD   0 /S$03
        BR   PR1
#18
        SB   1 /C7
        BR   PR1
        EM
```

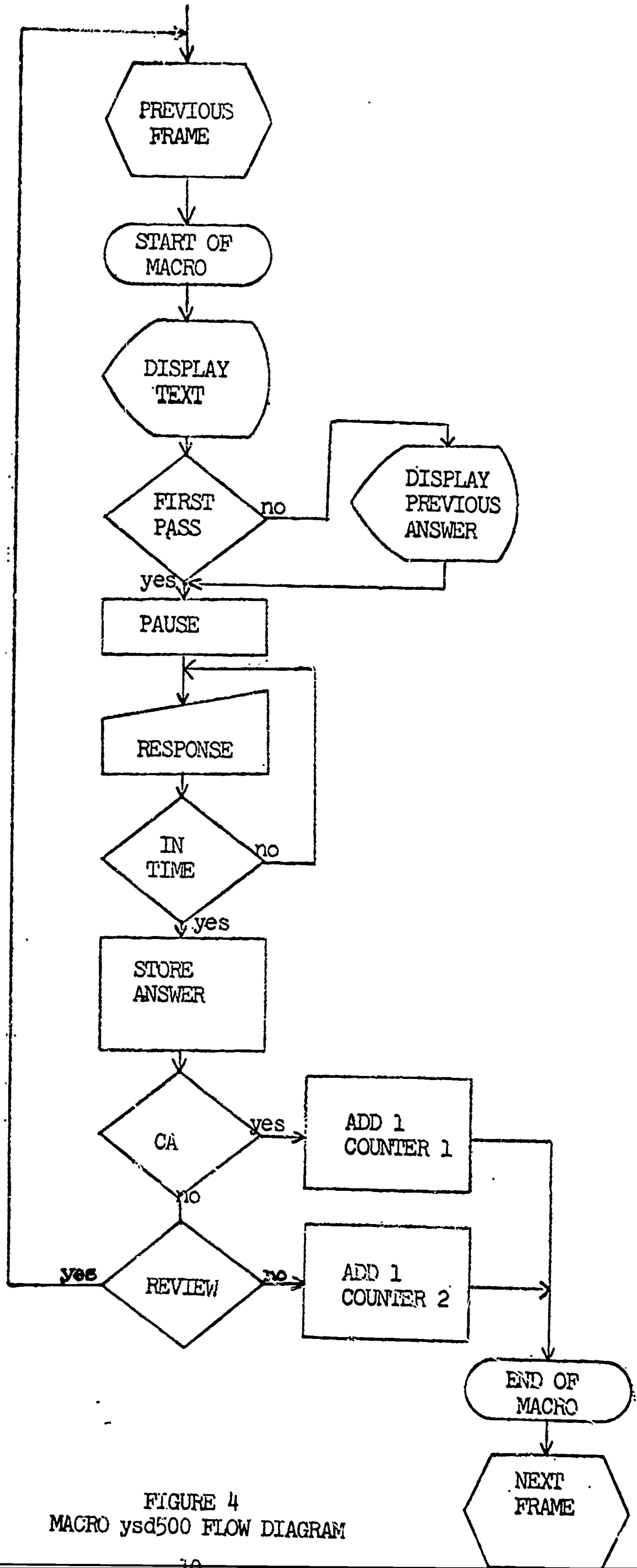


FIGURE 4
MACRO ysd500 FLOW DIAGRAM

ncd500
parameters
$01 review frame label
$02 switch and counter number for this frame
$03 switch and counter number for the next frame
$04 EPID

NOD macro expanded

```
          MA   NOD500 $     00000
          DTI  27,2 / /5,2 / / YES
          LR   $01 /RR4
          DTI  27,16 / /4,16 / / NO
          DTI  30,0 / /6,0 /$01
          DTI  27,31 / /8,31 / / REVIEW
          BR   #11 /C7 /G /9
          BR   #11 /S$02 /0
          BR   #12 /C$02 /E /2
          DTI  27,0 / /1,0 /X
          BR   #11
#12
          DTI  27,14 / /1,14 /X
#11
          PA   30
          EPP  9999 /$04
          NX
          BR   RE
          CAP  4,26,3,15 /CA
          LD   2 /C$02
          AD   1 /C1
          BR   #16
          AAP  4,26,3,30 /RV
          AD   1 /C3
          BR   $01
          WAP  4,26,3,1 /WY
          LD   1 /C$02
          AD   1 /C2
          BR   #16
          UN   UN
          BR   RE
#16
          BR   #18 /C7 /G /0
          LD   1 /S$02
          LD   0 /S$03
          BR   PR1
#18
          SB   1 /C7
          BR   PR1
          EM
```

Note: The flow diagram for this macro is identical to that shown in Figure 4 for ysd500.

mad500
parameters

$01	review frame label
$02	switch and counter number for this frame
$03	switch and counter number for the next frame
$04	EPID

MAD macro expanded

```
          MA   MAD500 $     00000
          DTI  14,32 / /6,32 /$01
          DTI  14,20 / /9,20 / / REVIEW
          LR   $01 /RR4
          DTI  16,1 / /3,1 /  /
          DTI  20,1 / /3,1 /  /
          DTI  24,1 / /3,1 /  /
          DTI  28,1 / /3,1 /  /
          BR   #31 /C7 /G /9
          BR   #31 /S$02 /0
          BR   #34 /C$02 /E /4
          BR   #33 /C$02 /E /3
          BR   #32 /C$02 /E /2
          DTI  16,0 / /1,0 /X
          BR   #31
#32
          DTI  20,0 / /1,0 /X
          BR   #31
#33
          DTI  24,0 / /1,0 /X
          BR   #31
#34
          DTI  28,0 / /1,0 /X
#31
          PA   30
          EPP  9999 /$04
          NX
          BR   RE
          CAP  4,15,3,1 /CA
          LD   1 /C$02
          AD   1 /C1
          BR   #36
          AAP  4,13,3,19 /RV
          AD   1 /C3
          AD   1 /C7
          BR   $01
          WAP  4,19,3,1 /WB
          LD   2 /C$02
          BR   #35
          WAP  4,23,3,1 /WC
          LD   3 /C$02
          BR   #35
          WAP  4,27,3,1 /WD
          BR   #35
          UN   UN
          BR   RE
```

(continued on next page)

macro mad500 continued

```
#35
     AD  1 /C2
#36
     BR  #38 /C7 /G /0

     LD  1 /S$02
     LD  0 /S$03
     BR  PR1
#38
     SB  1 /C7
     BR  PR1
```

Note: The flow diagram for this macro is identical to the one in Figure 4 for ysd500.

```
mbd500
parameters
$01                          review frame label
$02                          switch and counter number for this frame
$03                          switch and counter number for the next frame
$04                          EPID
```

MBD macro expanded

```
        MA   MBD500 $     00000
        LR   $01 /RR4
        DTI  14,32 / /6,32 /$01
        DTI  14,20 / /9,20 / / REVIEW
        DTI  16,1 / /3,1 /  /
        DTI  20,1 / /3,1 /  /
        DTI  24,1 / /3,1 /  /
        DTI  28,1 / /3,1 /  /
        BR   #31 /C7 /G /9
        BR   #31 /S$02 /0
        BR   #34 /C$02 /E /4
        BR   #33 /C$02 /E /3
        BR   #32 /C$02 /E /2
        DTI  16,0 / /1,0 /X
        BR   #31
#32
        DTI  20,0 / /1,0 /X
        BR   #31
#33
        DTI  24,0 / /1,0 /X
        BR   #31
#34
        DTI  28,0 / /1,0 /X
#31
        PA   30
        EPP  9999 /$04
        NX
        BR   RE
        CAP  4,19,3,1 /CA
        LD   2 /C$02
        AD   1 /C1
        BR   #36
        AAP  4,13,3,19 /RV
        AD   1 /C7
        AD   1 /C3
        BR   $01
        WAP  4,15,3,1 /WA
        LD   1 /C$02
        BR   #35
        WAP  4,23,3,1 /WC
        LD   3 /C$02
        BR   #35
        WAP  4,27,3,1 /WD
        LD   4 /C$02
        BR   #35
        UN   UN
        BR   RE
```

mbd500 continued

```
#35
     AD  1 /C2
#36
     BR  #38 /C27 /G /0
     LD  1 /S$02
     LD  0 /S$03
     BR  PR1
#38
     SB  1 /C7
     BR  PR1
     EN
```

Note: The flow diagram for this macro is identical to the one shown in Figure 4 for ysd500.

mcd500
parameters

$01	review frame label
$02	switch and counter number for this frame
$03	switch and counter number for the next frame
$04	EPID

MCD macro expanded

```
        MA   MCD500 $     00000
        DTI  14,32 / /6,32 /$01
        LR   $01 /RR4
        DTI  14,20 / /9,20 / / REVIEW
        DTI  16,1 / /3,1 /  /
        DTI  20,1 / /3,1 /  /
        DTI  24,1 / /3,1 /  /
        DTI  28,1 / /3,1 /  /
        BR   #31 /C7 /G /9
        BR . #31 /S$02 /0
        BR   #34 /C$02 /E /4
        BR   #33 /C$02 /E /3
        BR   #32 /C$02 /E /2
        DTI  16,0 / /1,0 /X
        BR   #31
#32
        DTI  20,0 / /1,0 /X
        BR   #31
#33
        DTI  24,0 / /1,0 /X
        BR   #31
#34
        DTI  28,0 / /1,0 /X
#31
        PA   30
        EPP  9999 /$04
        NX
        BR   RE
        CAP  4,23,3,1 /CA
        LD   3 /C$02
        AD   1 /C1
        BR   #36
        AAP  4,13,3,19 /RV
        AD   1 /C7
        AD   1 /C3
        BR   $01
        WAP  4,19,3,1 /WB
        LD   2 /C$02
        BR   #35
        WAP  4,15,3,1 /WA
        LD   1 /C$02
        BR   #35
        WAP  4,27,3,1 /WD
        LD   4 /C$02
        BR   #35
        UN   UN
        BR   RE
```

mcd500 continued

```
#35
     AD  1 /C2
#36
     BR  #38 /C7 /G /0
     LD  1 /S$02
     LD  0 /S$03

     BR  PR1
#38
     SB  1 /C7
     BR  PR1
     EM
```

Note: The flow diagram for this macro is identical to the one for ysd500 shown in Figure 4.

mdd500
parameters

$01	review frame label
$02	switch and counter number for this frame
$03	counter and switch number for the next frame
$04	EPID

MDD macro expanded

```
          MA   MDD500 $     00000
          DTI  14,32 / /6,32 /$01
          LR   $01 /RR4
          DTI  14,20 / /9,20 / / REVIEW
          DTI  16,1 / /3,1 /  /
          DTI  20,1 / /3,1 /  /
          DTI  24,1 / /3,1 /  /
          DTI  28,1 / /3,1 /  /
          BR   #31 /C7 /G /9
          BR   #31 /S$02 /0
          BR   #34 /C$02 /E /4
          BR   #33 /C$02 /E /3
          BR   #32 /C$02 /E /2
          DTI  16,0 / /1,0 /X
          BR   #31
#32
          DTI  20,0 / /1,0 /X
          BR   #31
#33
          DTI  24,0 / /1,0 /X
          BR   #31
#34
          DTI  28,0 / /1,0 /X
#31
          PA   30
          EPP  9999 /$04
          NX
          BR   RE
          CAP  4,27,3,1 /CA
          LD   4 /C$02
          AD   1 /C1
          BR   #36
          AAP  4,13,3,19 /RV
          AD   1 /C7
          AD   1 /C3
          BR   $01
          WAP  4,19,3,1 /WB
          LD   2 /C$02
          BR   #35
          WAP  4,15,3,1 /WA
          LD   1 /C$02
          BR   #35
          WAP  4,23,3,1 /WC
          LD   3 /C$02
          BR   #35
          UN   UN
          BR   RE
```

```
#35
     AD  1 /C2
#36
     BR  #38 /C7 /G /0
     LD  1 /S$02
     LD  0 /S$03

     BR  PR1
#38
     SB  1 /C7
     BR  PR1
     END
```

Note: The flow diagram for this macro is identical to the one for ysd500 shown in Figure 4.

rst500
parameters

$01	last frame label
$02	this frame label
$03	EPID

rst500 macro expanded

```
MA  RST500 $    00000
DTI 30,0 /2,30 /6,0 /$01
LR  $02 /RR2
DTI 27,29 /2,27 /8,29 / /REVIEW
DTI 27,3 /2,27 /10,3 / /CONTINUE
DTI 23,11 /2,23 /19,11 / /CALL FOR RESOURCE
PA  30
EPP 9999 /$03
NX
BR  RE
CAP 4,26,3,2 /TW
BR  PRI
AAP 4,22,3,10 /RS
BR  RESORS
AAP 4,26,3,28 /RV
BR  $01
UN  UN
BR  RE
EM
```

A listing is included below for the "Resors" statements which appeared at the end of each segment where it was called by rst500.

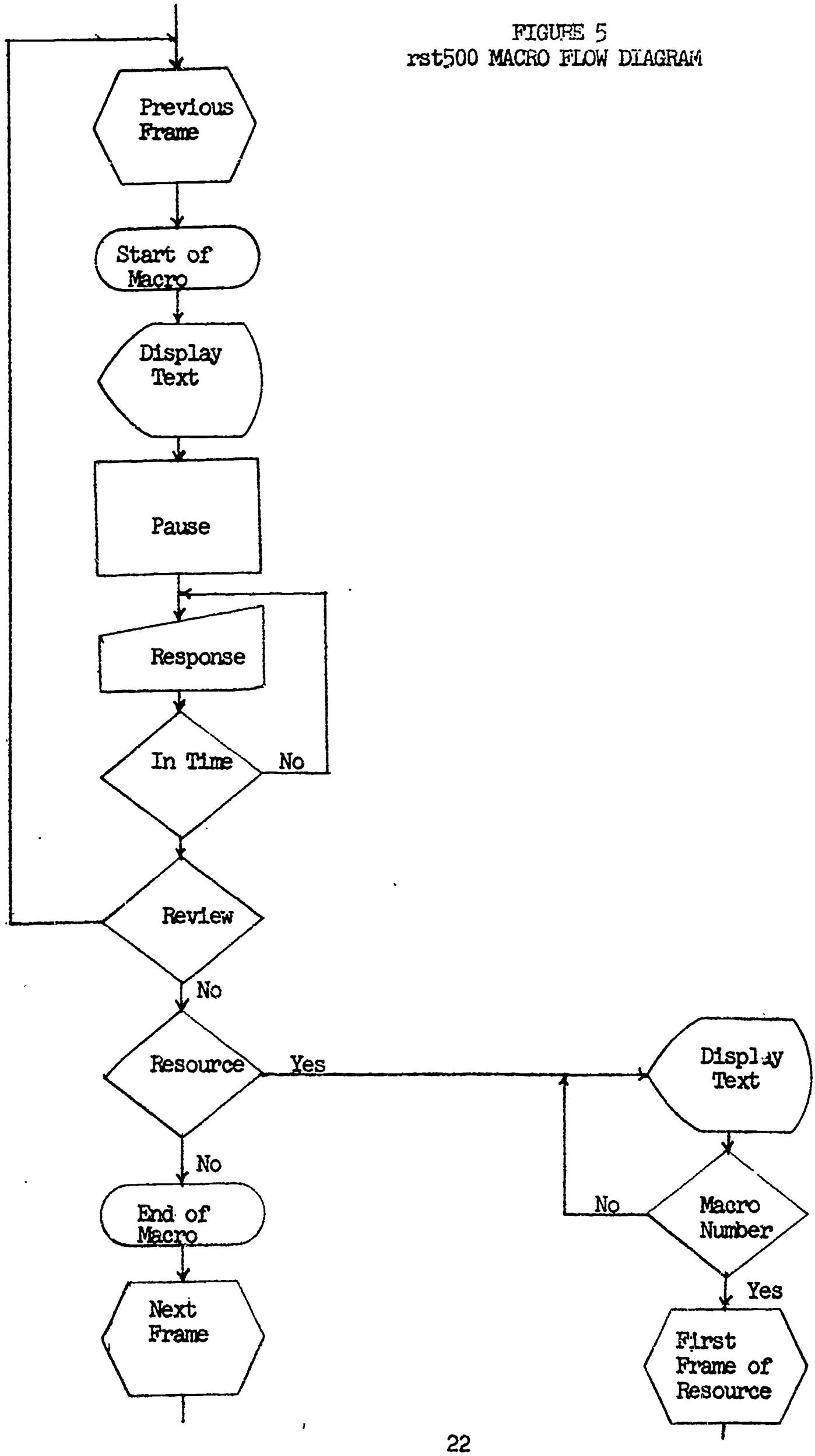
RESORS*E

```
 1  PRR *E
 2  DE  0+/32*E
 3  DT  0,0+/6,0+/40,0+/(U)SE THE (T)ABLE OF (C)ONTENTS IN YOUR *C*ISTUDENT TEXT
    TO SELECT THE (R)ESOURCE*C*IYOU WANT TO STUDY.*E
 4  DT 8,0+/4,8+/40,0+(TYPE THE NUMBER OF THE RESOURCE*C*IYOU HAVE SELECTED&*E
 5  EPI 10,27+/2,10+/3,27+/9999+/RESORS*E
 6  NX  *E
 7  BR  RE*E
 8  CA  1+/C1*E
 9  BR  1*E
10  CA  2+/C2*E
11  BR  2*E
 .
 .
 .
 .
 .
etc.
```

The following branch statements were used to transfer to the segment where the resources were located.

```
1*e
    1   epp 0>rcs1*e
    2   nx  *e
    3   tr  16>pa001*e
2*e
    1   epp 0>res2*e
    2   nx  *e
    3   tr  16>pb001*e
3*e
    1   epp 0>res3*e
    2   nx  *e
    3   tr  16>pc001*e
4*e
    1   epp 0>res4*e
    2   nx  *e
    3   tr  16>pd001*e
5*e
    1   epp 0>res5*e
    2   nx  *e
    3   tr  16>pe001*e
6*e
    1   epp 0>res6*e
    2   nx  *e
    3   tr  16>pf001*e
7*e
    1   epp 0>res7*e
    2   nx  *e
    3   tr  16>pg001*e
8*e
    1   epp 0>res8*e
    2   nx  *e
    3   tr  16>ph001*e
9*e
    1   epp 0>rcs9*e
    2   nx  *e
    3   tr  16>pi001*e
10*e
    1   epp 0>res10*e
    2   nx  *e
    3   tr  16>pj001*e
.
.
.
32*e
    1   epp 0>rs32*e
    2   nx  *e
    3   tr  16>pzf01*e
```

FIGURE 5
rst500 MACRO FLOW DIAGRAM



tst500*
parameters
$01 EPID
$02 Switch for correct answer
$03 Switch for first wrong answer
$04 Switch for second wrong answer
$05 Switch for third wrong answer
$06 First line (on the CRT screen) of correct answer response
$07 Letter representing position of the correct answer (A, B, C, or D)
$08 First line of the first wrong answer
$09 Letter for position of first wrong answer
$10 First line of second wrong answer
$11 Letter for position of second wrong
$12 First line of third wrong answer
$13 Letter for position of third wrong answer
$14 Previous frame label

*Note: This version of tst500 is a revision of the original macro which was used to program the text for the test for Volume 3A. This macro was used to program the text for one-half of Volume 3B.

(The expanded macro is on the following page.)

```
macro tst500 expanded

  1  DTI 30,24+/2,30+/6,24+/+/ SKIP*E
  2  DTI 30,31+/2,30+/8,31+/+/ REVIEW*E
  3  DTI 14,0+/2,14+/3,0+/ +/ *E
  4  DTI 18,0+/2,18+/3,0+/ +/ *E
  5  DTI 22,0+/2,22+/3,0+/ +/ *E
  6  DTI 26,0+/2,26+/3,0+/ +/ *E
  7  LR  (3)11(3)45+/RR4*E
  8  BR  (3)22(3)45+/S$02+/1*E
  9  BR  (3)33(3)45+/S$03+/1*E
 10  BR  (3)44(3)45+/S$04+/1*E
 11  BR  (3)55(3)45+/S$05+/1*E
(3)11(3)45*E
  1  EPP 9999+/$01*E
  2  NX  *E
  3  BR  RE*E
  4  CAP 4,$06,3,0+/C$07*E
  5  AD  1+/C25*E
  6  LD  1+/S$02*E
  7  BR  PR1*E
  8  WAP 4,$08,3,0+/W$09*E
  9  AD  1+/C26*E
 10  LD  1+/S$03*E
 11  BR  PR1*E
 12  WAP 4,$10,3,0+/W$11*E
 13  AD  1+/C26*E
 14  LD  1+/S$04*E
 15  BR  PR1*E
 16  WAP 4,$12,3,0+/W$13*E
 17  AD  1+/C26*E
 18  LD  1+/S$05*E
 19  BR  PR1*E
 20  AAP 3,39,2,23+/SK*E
 21  AD  1+/C28*E
 22  BR  PR1*E
 23  AAP 3,29,2,20+/RV*E
 24  AD  1+/C27*E
 25  BR  S$14*E
 26  UN  UN*E
 27  BR  RE*E
(3)22(3)45*E
  1  DTI 14,0+/2,14+/3,0+/
  2  BR  RR4*E
(3)33(3)45*E
  1  DTI 18,0=/2,18+/3,0+/
  2  BR  BR4*E
(3)44(3)45*E
  1  DTI 22,0+/2,22+/3,0+/
  2  BR  RR4*E
(3)55(3)45*E
  1  DTI 26,0+/2,26+/3,0+/
  3  BR  RR4*E
```

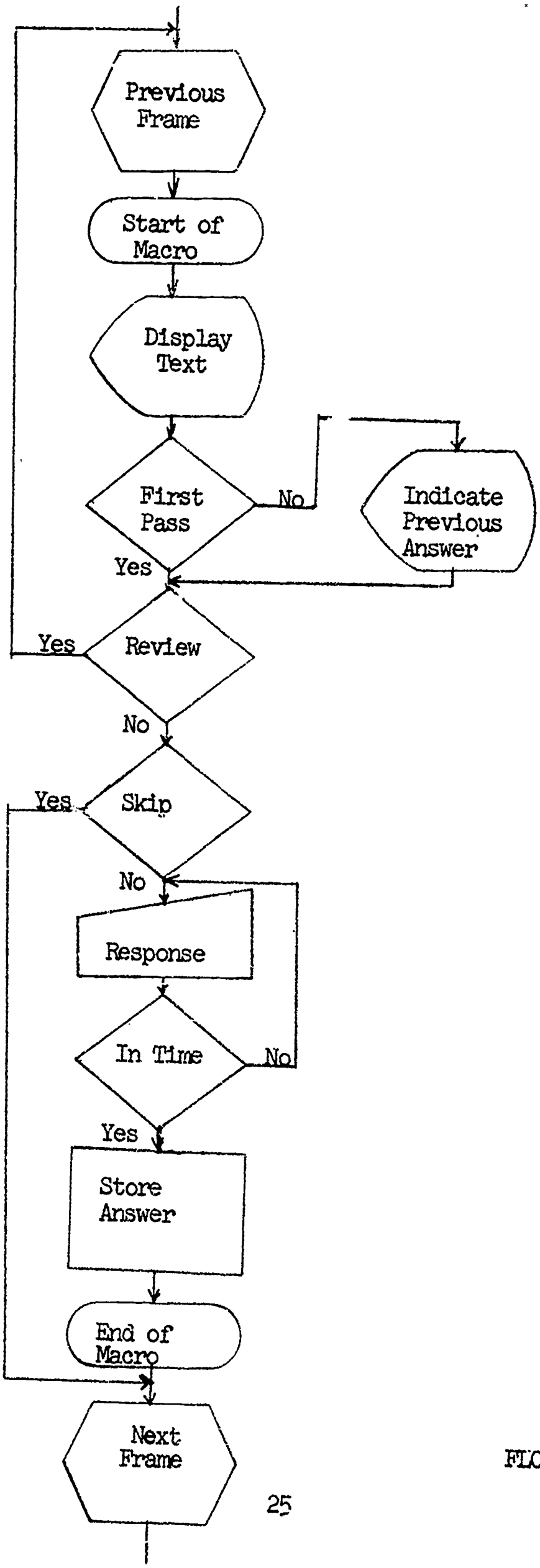


FIGURE 6
FLOW DIAGRAM FOR tst500

dcd500
parameters

$01	review frame label
$02	buffer number for this frame
$03	counter and switch number for this frame
$04	counter and switch number for the next frame
$05	EPID

DCD macro expanded

```
        MA   DCD500 $     00000
        DTI  30,0 / /6,0 /$01
        DT   26,5 / / / I F YOU WISH TO REVIEW, TYPE XX.
        LR   $01 /RR4
        BR   #11 /C7 /G /3
        BR   #11 /S$03 /0
        DT   21 / / / Y OUR ANSWER LAST TIME WAS
        DT   24 / / /B$02
        DT   28 / / / C ONTINUE BY REANSWERING THE QUESTION.
#11
        PA   30
        EP   17 / / /9999 /40 /$05
        NX
        BR   RE
        AA   XX /RV
        AD   1 /C7
        AD   1 /C3
        BR   $01
        CA    8   /DC
        LD   B0 /B$02
        BR   #44 /C7 /G /0
        LD   1 /S$03
        LD   0 /S$04
        BR   PR1
#44
        SB   1 /C7
        BR   PR1
        EM
```

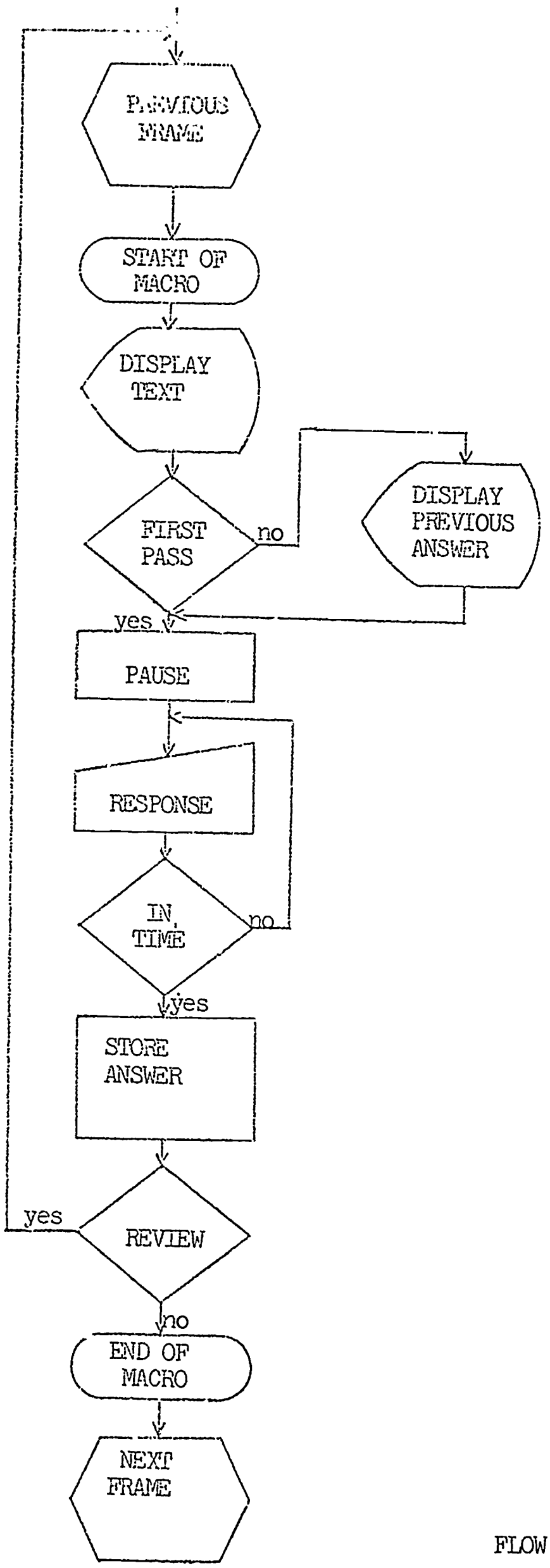


FIGURE 7
FLOW DIAGRAM FOR dcd500

cmt500
parameters

$01	previous frame label
$02	EPID
$03	next frame label

macro cmt500 expanded

```
MA  CMT500 $     00000
DTI 30,0 / /6,0 /$01
DTI 27,26 / /8,26 / / REVIEW
DTI 27,3 / /19,3 / / READY TO COMMENT
PA  30
EPP 9999 /$02
NX
BR  RE
CAP 4,26,3,2 /CT
LR  $03 /RR5
BR  HOLD
AAP 4,26,3,25 /RV
BR  $01
UN  UU
BR  RE
EM
```

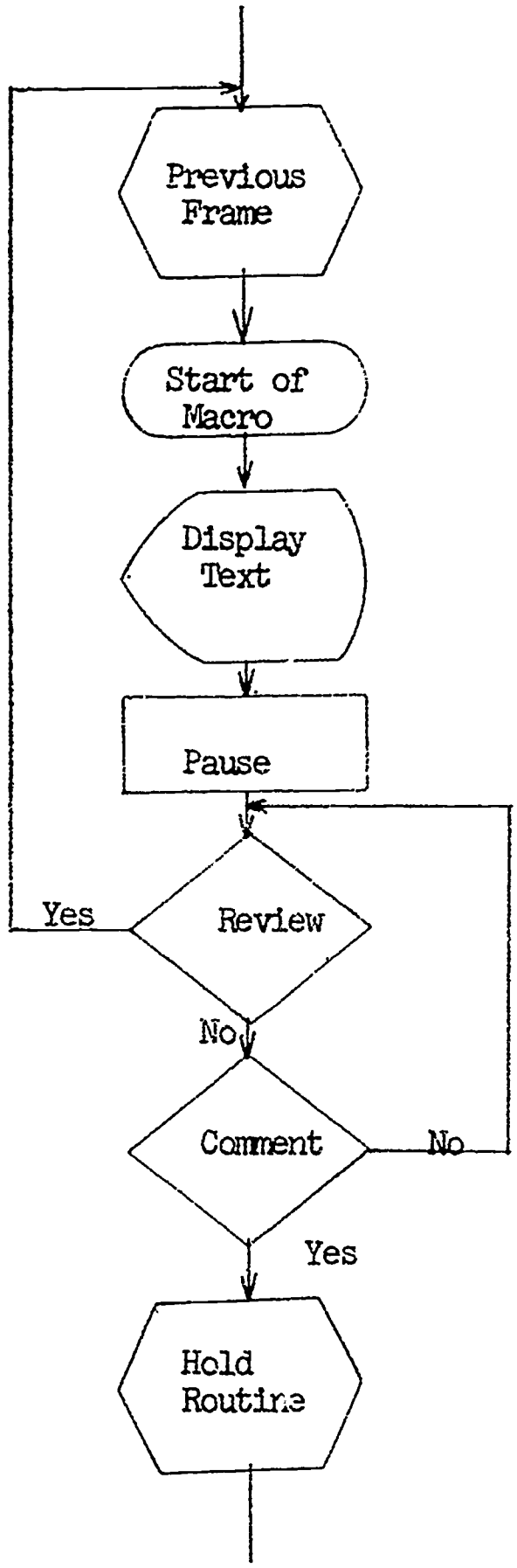


FIGURE 8
FLOW DIAGRAM FOR cmt500

lmd500
parameters

$01	review frame label
$02	buffer number for this frame
$03	switch and counter number for this frame
$04	switch and counter number for the next frame
$05	EPID
$06	lower numerical limit
$07	upper numerical limit

LMD macro expanded

```
        MA   LMD500 $     00000
        DTI  30,0 / /6,0 /$01
        DT   26,5 / / / I F YOU WISH TO REVIEW, TYPE XX.
        LR   $01 /RR4
        BR   #11 /C7 /G /3
        BR   #11 /S$03 /0
        DT   21 / / / Y OUR ANSWER LAST TIME WAS
        DT   24 / / /B$02
        DT   28 / / / C ONTINUE BY REAN$WERING THE QUESTION.
#11
        PA   30
        EP   17 / / /9999 /40 /$05
        NX
        BR   RE
        AA   XX /RV
        AD   1. /C7
        AD   1 /C3
        BR   $01
        FN2  LT / / /$06 /$07 /C /CF
        AD   1 /C1
        BR   #33
        WA    8  /WF
        AD   1 /C2
#33
        LD   80 /B$02
        BR   #44 /C7 /G /0
        LD   1 /S$03
        LD   0 /S$04
        BR   PR1
#44
        SB   1 /C7
        BR   PR1
        EM
```

Note: The flow diagram for lmd500 is identical to the one shown in Figure 4 above.

kld500
parameters

$01	review frame label
$02	buffer number for this frame
$03	switch and counter number for this frame
$04	switch and counter number for the next frame
$05	EPID
$06	correct answer
$07	number of words necessary for a correct match

KLD macro expanded

```
MACWTR
        MA   KLD500 $     00000
        DTI  30,0 / /6,0 /$01
        DT   26,5 / / / I F YOU WISH TO REVIEW, TYPE XX.
        LD   $06 /B4
        LR   $01 /RR4
        BR   #11 /C7 /G /3
        BR   #11 /S$03 /0
        DT   21 / / / Y OUR ANSWER LAST TIME WAS
        DT   24 / / /B$02
        DT   28 / / / C ONTINUE BY REANSWERING THE QUESTION.
#11
        PA   30
        EP   17 / / /9999 /40 /$05
        NX
        BR   RE
        AA   XX /RV
        AD   1 /C7
        AD   1 /C3
        BR   $01
        FN2  KEYL /$07 /C / /CF
        AD   1 /C1
        BR   #33
        WA    8  /WF
        AD   1 /C2
#33
        LD   B0 /B$02
        BR   #44 /C7 /G /0
        LD   1 /S$03
        LD   0 /S$04
        BR   PR1
#44
        SB   1 /C7
        BR   PR1
        EM
```

Note: The flow diagram for kld500 is identical to Figure 4 above.

ISCS MACRO RESPONSE IDENTIFIER SCHEME

For frames which contain a question for the student to answer.

multiple choice answer

- CA correct response
- WA incorrect response, student picked choice A
- WB incorrect response, student picked choice B
- WC incorrect response, student picked choice C
- WD incorrect response, student picked choice D

yes - no answer

- CA correct response
- WN incorrect response, student picked "no," "yes" was correct
- WY incorrect response, student picked "yes," "no" was correct

free response question

- CF correct response, through the use of the limit or keyletter function
- WF incorrect response, through the use of the limit or keyletter function
- DC "don't care" response, which must be examined manually if a correct or incorrect interpretation is necessary

The above response identifiers are the only ones which are used in the macro system to detect correct or incorrect responses. Note that all correct responses have a C in the first position of the identifier, and all incorrect responses have a W in the first position. (C = correct, W = wrong) The second position is a modifier which allows you to determine the type of question.

Other identifiers used, which are not associated with a correct or incorrect response:

- RV review
- UN unrecognizable response
- EX selection of an excursion
- TM continue, under macro ttd500
- OT time out, a identifier put in by the system
- MM answer mismatch, put in by the system
- CC comment response indicating that a comment will be entered by the student

The Ten-Digit ISCS EPID Code

The ten-digit ISCS EPID code included eight fields as shown in the diagram below:

Chapter	Page	Content	Process	Track	Question Code	Question Number	Sequence Number
1	2	3	4	5	6	7-8	9-10

Descriptions of each field

1. Chapter Alphanumeric code, a-z, corresponding to chapter numbers 1-26.

2. Page Alphanumeric code, a-z, corresponding to page numbers 1-26.

3. Content Alphanumeric code, a-z and 1-9, for identification of the frame.

4. Process Alphanumeric code a-z, 1-9, for identification of the scientific process involved in the frame. The representation is the same for both seventh and eighth grades.

5. Track Alphanumeric code, corresponding to the following key:
   m mainline
   e, i, x, excursion for first, second, or third set of 26 excursions respectively.

6. Question Code Each frame contains an alphanumeric code to allow for sorting on questions.

   a no question in frame, only the presentation of information
   x question in the computer program to be answered at the terminal
   t, y, z, question, with answer to be written in student text

7-8. Question Number Two character numeric code, which corresponds to the question number in the classroom text. If there is no question, the code is 00.

Note: If the frame in quest was an excursion decision frame, EPID characters 3 and 4 were used for the chapter number and characters 7 and 8 were used for the excursion number.

9-10. Sequence Number Two character numeric code, 01-99, corresponding to the frame sequence number. The first frame of each segment is 01, the next 02, etc. If there are more than 99 frames, the code recycles, starting with 01.

Content and Process Codes

The content and process code designations used for the grades seven and eight programs are defined in the lists below. It should be noted that separate lists are used for grades seven and eight content, while a common list was used for the processes. As of the date of this publication, no separate content list is available for the ninth grade materials.

Content Alphanumeric code, a →z and 1 →9, for identification of the scientific content of the frame.

Code	Grade 7	Grade 8
a	models	definition
b	force	chemical nomenclature
c	friction	word statements
d	distance	chemical reaction
e	work	chemical test
f	time	chemical system
g	speed	model, characteristics of models
h	potential energy	particle model
i	kinetic energy	atoms, elements
j	chemical energy	ions
k	electrical energy	molecules
l	light energy	structure
m	heat energy	compounds
n	energy in general	conservation of matter
o	temperature	properties of matter
p	energy conversion	mass-weight
q	weight-mass	math, graphing
r	momentum	combination & permutations
s	resistance	volume
t	voltage	reaction time, time
u	current	energy, work, temp. change
v	circuitry	electrical charge
w	calorimetry	forces
x	phase change	combining power
y	expansion	density
z	particles	concentration
1	establishing standards	catalysts
2	math	sciencing
3	quantification	chemical composition
4	definition	
5	sciencing	
6	moments	
7	chemical change	
8	magnetism	
9	miscellaneous	miscellaneous

Process — Alphanumeric code a → z, 1 → 9, for identification of the scientific process involved in the frame. The representation is the same for both seventh and eighth grades.

a	functional definition
b	operational definition
c	establishing standards
d	choice and use of correct units
e	use of instruments
f	making observations
g	averaging
h	numeric operations
j	arranging axis and plotting points
k	interpreting graphical data
l	interpreting non-graphical data
m	specific factual recall
n	identifying variables
o	controlling variable
p	hypothesis formation
q	hypothesis testing
r	model definition and construction
s	application of course models
u	application of a principle or concept
v	equipment instructions
w	statement of a principle or concent
x	thought stimulation or focus
y	information presentation
z	answer to previous question
1	filling a data table
2	review of previous material
3	systems analysis
4	descriptive definition
5	predicting
9	garbage

ISCS LABEL SCHEME

Each frame programmed by ISCS was labeled by means of the following scheme. The label served the usual Coursewriter II functions in regard to branching, restart, etc.

6 Characters

1	2	3	4	5	6

1. Chapter Alphanumeric code a→z, corresponding to chapter number 1 →26.

2. Page Number Alphanumeric code a→z, corresponding to page number 1→ 26.

3-4-5. Sequence Number Numeric code, 0→900 corresponding to the frame sequence number. The first frame of each course segment begins with the number 1, and they are numbered consecutively from that point.

6. Extra This position is normally blank. If frames must be inserted at a later date, this position allows for keeping the same label scheme, and using an alphanumeric code in this position to distinguish between frames.

The Analysis Programs

There were five analysis programs developed by ISCS for processing the data that was recorded, sorted, and merged by the CAI Center's Data Management System (DMS) operating under the ISS Monitor. In addition, the center provided item analysis summaries and listings of student comments. The latter two analysis were performed by the center, using the sort and analysis capability of the DMS. Any inquiries relative to the DMS-ISS monitor system or operations should be directed to Mr. George Hogshead, CAI Center, Florida State University, Tallahassee, Florida 32306.

Detailed Listings and Documentation

Program logic description, operator instructions, flow diagrams, and detailed program listings with comments are presented for each of the five ISCS analysis programs.

PROGRAM ISCSA

PROGRAM LOGIC DESCRIPTION

The purpose of the program is to provide a detailed, but edited, listing of student responses. The printed records consist of the student number, EPID, response identifier, date of response, latency, and the contents of buffer zero.

The program runs on a work tape, in regular nistory file sequence, containing only the student population wanted. The tape may be mounted on either drive.

1. The program reads a student record, using subroutine FAKE, into the 134 character array IREC*. If it is a new student, the program goes to 2 below. If it is not a new student, the program goes to 3 below.

2. The program stores the student number, skips to a new page on the printer, and writes the page header.

3. The program prints the student number, EPID, response identifier, latency, date of response, and buffer zero, then returns to 1 above.

After each read statement, an end of file check is made.

*The layout for array IREC is given in the Appendix

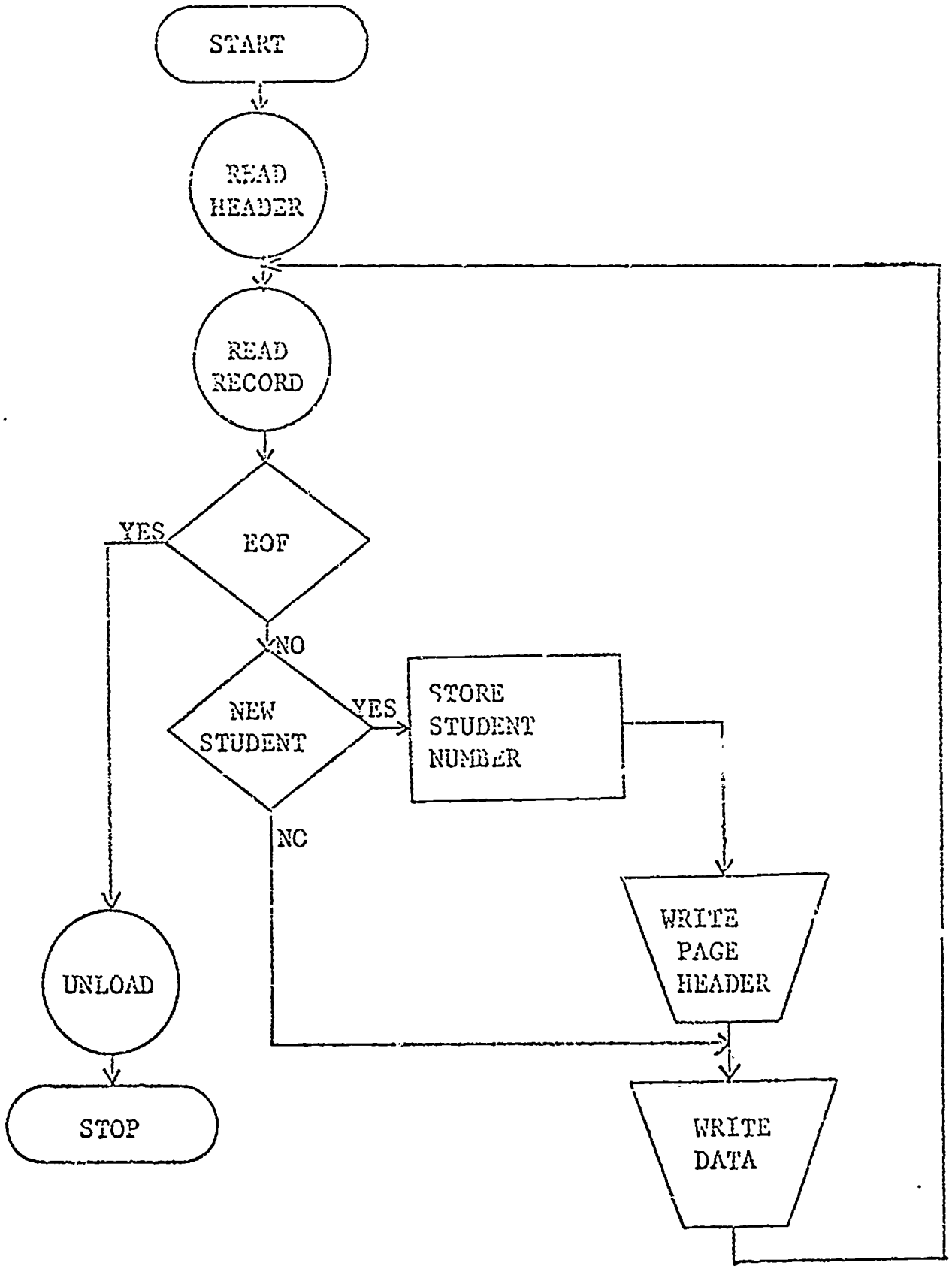


FIGURE 9
PROGRAM ISCSA FLOW DIAGRAM

```
C                                                                       ISA00050
C                                                                       ISA00060
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * ISA00070
C                                                                       ISA00080
C                    THIS PROGRAM WAS WRITTEN BY                        ISA00090
C                          DAVID H DASENBROCK                           ISA00100
C                          NOVEMBER, 1968
C                                                                       ISA00110
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * ISA00120
C                                                                       ISA00130
C   THE PURPOSE OF THIS PROGRAM IS TO PROVIDE AN EDITED DETAILED LISTING A00140
C  WHICH IS MORE COMPACT THAN THAT PROVIDED BY THE CENTER.  THE OUTPUT LISTS 150
C  THE STUDENT NUMBER, DATE, LATENCY, BUFFER 0, EPID, AND RESPONSE IDENTIFIER 60
C  THE PRIMARY USE OF THIS PROGRAM IS FOR REFERENCE.  THE PROGRAM RUNS OFF OF 70
C  TAPE OF THE STUDENTS IN HISTORY FILE SEQUENCE                        ISA00180
                                                                        ISA00190
C                                                                       ISA00200
      COMMON IREC(134),JDATE(4)                                         ISA00210
C        IREC = 134 CHARACTER ARRAY FOR THE TAPE RECORD
      DATA IA,IB,NUM/'F ','C ','XX'/                                    ISA00220
      TOTAL=0.0                                                         ISA00230
      N=2                                                               ISA00240
C  SET TAPE DRIVE SENSE SWITCH                                          ISA00250
      WRITE(2,3004)                                                     ISA00260
3004  FORMAT(//,20X,'SET TAPE DRIVE SENSE SWITCH',/,                    ISA00270
     *20X,' SS 4 ON = USE TAPE DRIVE 5',/,                              ISA00280
     *20X,' SS 4 OFF= USE TAPE DRIVE 6',///////)                        ISA00290
      PAUSE                                                             ISA00300
      CALL SSWTCH(4,JD)                                                 ISA00310
      GO TO (3001,3002),JD                                              ISA00320
3001  IDRIV=5                                                           ISA00330
      GO TO 19                                                          ISA00340
3002  IDRIV=6                                                           ISA00350
C  READ THE HEADER                                                      ISA00360
19    CALL RHDR(IDRIV)                                                  ISA00370
C  READ A STUDENT RECORD                                                ISA00380
C  A STUDENT RECORD IS READ INTO THE IREC ARRAY USING SUBROUTINE FAKE.
20    CALL FAKE(IDRIV,N)                                                ISA00390
C  CHECK FOR END OF FILE                                                ISA00400
      IF(N-2)50,10,50                                                   ISA00410
C  CHECK FOR A NEW STUDENT                                              ISA00420
10    IF(IREC(5)-NUM)11,13,11                                           ISA00430
11    CONTINUE                                                          ISA00440
      WRITE(2,111)  TOTAL                                               ISA00450
111   FORMAT(//,5X,'TOTAL TIME IN CHAPTER  ',F6.1,'  MINUTES')          ISA00460
      TOTAL=0.0                                                         ISA00470
C  WRITE NEW PAGE HEADER                                                ISA00480
      WRITE(2,12)                                                       ISA00490
12    FORMAT(1H1,////,40X,'STUDENT RESPONSE PROFILE',//)                ISA00500
      NUM=IREC(5)                                                       ISA00510
13    CALL DATE                                                         ISA00520
      TIME=IREC(22)/600.0                                               ISA00530
      TOTAL=TOTAL+TIME                                                  ISA00540
C      ADD TIME IN MINUTES                                              ISA00550
```

```
      IF(IREC(21)-IA)14,16,14                                          ISA00560
14    IF(IREC(21)-IB)15,16,15                                          ISA00570
C     WRITE RECORD                                                     ISA00580
15    WRITE(2,17)IREC(4),IREC(5),(IREC(J),J=10,21),JDATE,TIME          ISA00590
17    FORMAT(2X,2A2,2X,10A1,2X,2A1,2X,4A2,2X,F6.2)                     ISA00600
18    FORMAT(2X,2A2,2X,10A1,2X,2A1,2X,4A2,2X,F6.2,2X,22A2)             ISA00610
      GO TO 20                                                         ISA00620
16    WRITE(2,18)IREC(4),IREC(5),(IREC(J),J=10,21),JDATE,TIME,         ISA00630
     *(IREC(K),K=84,104)                                               ISA00640
      GO TO 20                                                         ISA00650
50    REWIND IDRIV                                                     ISA00660
      STOP                                                             ISA00670
      END                                                              ISA00680
// XEQ ISCSA                                                           ISA00690
```

PROGRAM ISCSA

OPERATING INSTRUCTIONS

The program is designed to print an edited detailed listing, from a work tape in history file sequence, of the selected ISCS students.

To run the program

1. mount the work tape (either drive)
2. put stock paper in printer
3. load fortran deck
4. set sense switch 4 to assign the proper tape drive. (drive selection instructions appear on printer)

OUTPUT

1. tape drive selection instruction
2. student listing

PROGRAM ISCSP

PROGRAM LOGIC DESCRIPTION

The purpose of this program is to provide a graphic plot of each student's progress through the instructional materials, with respect to time. The ISCS-CAI macro system is such, that the last two characters of the EPID contain the frame sequence number. The frames are numbered consecutively, starting with 1 at the beginning of each course segment. The program uses this sequence number for reference when plotting the data points.

The program runs on a work tape, in regular history file sequence, containing only the student population wanted. The tape may be mounted on either tape drive.

1. The program reads a student record, using subroutine FAMB, into the 134 character array IREC. If it is a new student, the program goes to 2 below. If it is not a new student, the program goes to 3 below.

2. The program stores the student number, skips to a new page on the printer, and writes the page header, which includes the student number, and x axis of the graph.

3. The program examines IREC(18) and IREC(19), which are the last two characters of the EPID. The numerical value of these two positions are found, K=(10 x IREC(18) + IREC(19), and a x placed as the data point in Line (K).

4. The latency from the record is added to the total latency. If the total latency is less than 5 minutes, the program returns to 1.

5. If the total latency is greater than 5, 5 is subtracted from the total latency, array LINE is printed, then set equal to blanks, except the last data point. The program returns to 1 above.

After each read statement, an end of file check is made.

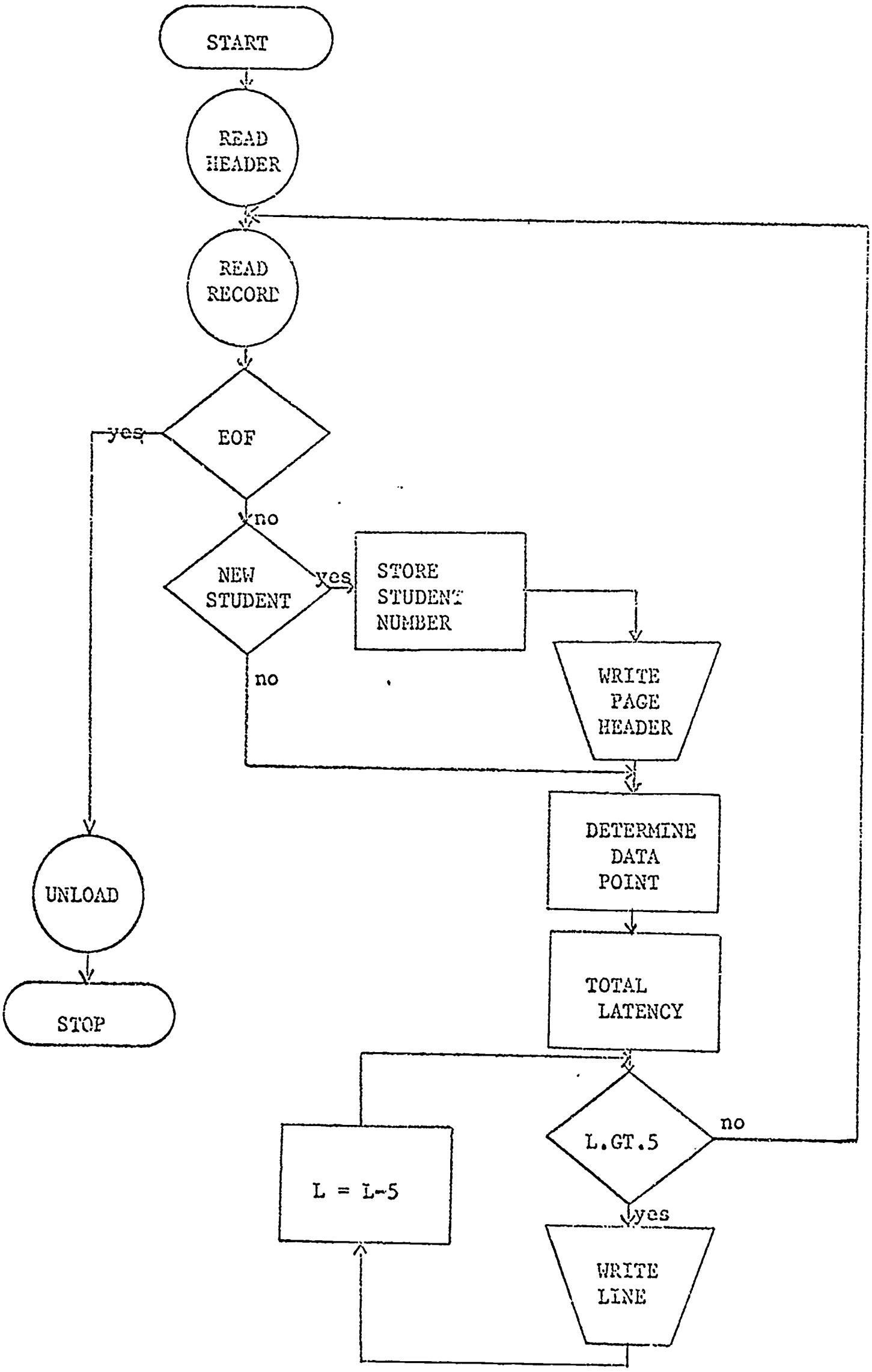


FIGURE 10
PROGRAM ISCSP FLOW DIAGRAM

```
      DIMENSION NUM(10),LINE(102)                                        ISP00050
      COMMON IREC(134),JDATE(4)                                          ISP00060
C                                                                        ISP00070
C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  ISP00080
C                                                                        ISP00090
C                    THIS PROGRAM WAS WRITTEN BY                         ISP00100
C                           DAVID H DASENBROCK                           ISP00110
C                           JANUARY, 1969                                ISP00120
C                                                                        ISP00130
C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  ISP00140
C     THE PURPOSE OF THIS PROGRAM IS TO PLOT STUDENT PROGRESS THROUGH THE ISCS 50
C   INSTRUCTIONAL MATERIALS WITH RESPECT TO TIME.  THE PROGRAM RUNS OFF OF A WORK
C   TAPE, WITH THE STUDENTS IN HISTORY FILE ORDER.  THE PROGRAM PLOTS TIME IN 170
C   FIVE MINUTE INTERVALS AS THE Y AXIS, AND THE FRAME SEQUENCE NUMBER AS THE X 0
C   AXIS.  THE FRAM SEQUENCE NUMBER IS DERIVED FROM THE LAST TWO CHARACTERS 00190
C   OF THE EPID.                                                         ISP00200
C   NUM= ALPHANUMERIC ARRAY CONTAINING 0-9M FOR DETERMINING FRAME NUMBER ISP00210
C   LINE = AP                                                            ISP00220
C   LINE = ALPHANUMERIC ARRAY FOR PRINTING LINE OF PLOT                  ISP00230
      DATA NUM/'0 ','1 ','2 ','3 ','4 ','5 ','6 ','7 ','8 ','9 '/       ISP00240
      DATA IX,IY,IBLNK,IDOT,ID,N/'X','-',' ','.','40',2/                 ISP00250
C   DETERMINE TAPE DRIVE                                                 ISP00260
C SS2 ON, JQ=1, QUESTIONS ONLY/ SS3 ON, JP=1, PUNCH CARDS                ISP00270
      WRITE(2,3004)                                                      ISP00280
3004  FORMAT(//,20X,'SET TAPE DRIVE SENSE SWITCH',/,                     ISP00290
     *20X,' SS 4 ON = USE TAPE DRIVE 5',/,                               ISP00300
     *20X,' SS 4 OFF= USE TAPE DRIVE 6',///////)                         ISP00310
      PAUSE                                                              ISP00320
      CALL SSWTCH(4,JD)                                                  ISP00330
      GO TO (3001,3002),JD                                               ISP00340
3001  IDRIV=5                                                            ISP00350
      GO TO 3003                                                         ISP00360
3002  IDRIV=6                                                            ISP00370
C   READ HEADER                                                          ISP00380
3003  CALL RHDR(IDRIV)                                                   ISP00390
C   READ RECORD                                                          ISP00400
10    CALL FAKE(IDRIV,N)                                                 ISP00410
C   CHECK FOR END OF FILE                                                ISP00420
      IF(N-2)5,11,5                                                      ISP00430
C   CHECK FOR NEW STUDENT                                                ISP00440
C STORE NEW  STUDENT NUMBER , TERMINATE PLOT, GO TO NEW PAGE, AND START
C THE NEXT PLOT.
C IF NOT A NEW STUDENT, GO TO WHERE POSITION OF DATA POINT IS DETERMINED
C AND A CHECK IS MADE FOR 5 MINUTE INTERVAL.
11    IF(ID-IREC(5))12,40,12                                             ISP00450
12    ID=IREC(5)                                                         ISP00460
      MA=J*5                                                             ISP00470
      WRITE(2,102)MA,LINE                                                ISP00480
100    WRITE(2,101)                                                      ISP00490
C   WRITE PAGE HEADER                                                    ISP00500
101   FORMAT(1H1)                                                        ISP00510
      WRITE(2,120)IREC(1),IREC(2),IREC(3)                                ISP00520
120   FORMAT(50X,3A2)                                                    ISP00530
```

```
      WRITE(2,110)IREC(4),IREC(5) ,IREC(10)                              ISP00540
110   FORMAT(35X,'STUDENT NUMBER  ',2A2,10X,'CHAPTER ',A2,/)             ISP00550
      WRITE(2,106)                                                       ISP00560
106   FORMAT(2X,'TIME',45X,'FRAME')                                      ISP00570
      WRITE(2,107)                                                       ISP00580
107   FORMAT(1X,'MINUTES')                                               ISP00590
      WRITE(2,108)                                                       ISP00600
      WRITE(2,105)                                                       ISP00610
C SET UP THE LEGEND FOR THE PLOT.
108   FORMAT(19X,'1111111111222222222233333333334444444444555555555566661SP00620
     *66666677777777778888888888999999999')                              ISP00630
105   FORMAT(10X,'12345678901234567890123456789012345678901234567890123ISP00640
     *567890123456789012345678901234567890123456789')                    ISP00650
      L=0                                                                ISP00660
C  SET AXIS                                                            ISP00670
      DO 13 I=1,102                                                      ISP00680
      LINE(I)=IDOT                                                       ISP00690
13    CONTINUE                                                           ISP00700
      J=0                                                                ISP00710
      WRITE(2,102)J,LINE                                                 ISP00720
102   FORMAT(2X,I5,2X,102A1)                                             ISP00730
60    DO 14 I=2,101                                                      ISP00740
      LINE(I)=IBLNK                                                      ISP00750
14    CONTINUE                                                           ISP00760
      DO 15 I=1,9                                                        ISP00770
      M=I*10+1                                                           ISP00780
      LINE(M)=IY                                                         ISP00790
   15 CONTINUE                                                           ISP00800
      J=J+1                                                              ISP00810
      K=(J/10)*10                                                        ISP00820
      IF(J-K)40,20,40                                                    ISP00830
20    DO 21 I=2,101,10                                                   ISP00840
      MOP=I+4                                                            ISP00850
      LINE(MOP)=IY                                                       ISP00860
      MPO=I+8                                                            ISP00870
      LINE(MPO)=IY                                                       ISP00880
      LINE(I)=IY                                                         ISP00890
21    CONTINUE                                                           ISP00900
C  DETERMINE COORDINATES OF DATA POINT                                 ISP00910
40    DO 45 I=1,10                                                       ISP00920
      IF(IREC(18)-NUM(I))43,42,43                                        ISP00930
42    IA=I-1                                                             ISP00940
43    IF(IREC(19)-NUM(I)) 45,44,45                                       ISP00950
44    IB=I-1                                                             ISP00960
45    CONTINUE                                                           ISP00970
      IF(IA-0)47,46,47                                                   ISP00980
46    IF(IB-0)47,10,47                                                   ISP00990
47    IC=IA*10+IB+1                                                      ISP01000
C TOTAL TIME IS L.
      LINE(IC)=IX                                                        ISP01010
      L=L+IREC(22)                                                       ISP01020
      IF(L-3000)10,50,50                                                 ISP01030
C  WRITE LINE                                                          ISP01040
C  E DATA POINT IS PLOTTED AS A LETTER...LINES ARE PLOTTED AS DOTS WITH
```

```
C DASHES FOR FIVE MINUTE INTERVALS.
50      MA=J*5                                                          ISP01050
        WRITE(2,102)MA,LINE                                             ISP01060
        L=L-3000-IREC(22)                                               ISP01070
        GO TO 60                                                        ISP01080
C   UNLOAD THE BLOODY TAPE, THE PROGRAM HAS RUN OUT OF THE STUPID STUFF ISP01090
5       REWIND IDRIV                                                    ISP01100
        STOP                                                            ISP01110
        END                                                             ISP01120
// XEQ ISCSP                                                            ISP01130
```

PROGRAM ISCSP

OPERATING INSTRUCTIONS

This program is designed to read a work tape, comprised of ISCS students in history file sequence, and provide a plot of student progress, with respect to 5 minute time intervals.

To run the program

1. mount work tape (either drive)
2. put stock paper in printer
3. load fortran deck
4. set sense switch 4 to assign proper tape drive. (drive selection instructions appear on printer)

OUTPUT

1. tape drive selection instructions
2. plot of students path through the instructional materials, with respect to time.

PROGRAM ISCST

PROGRAM LOGIC DESCRIPTION

The purpose of this program is to provide a graphic trace of each student's progress through the instructional sequence. The ISCS-CAI macro system is such that the last two characters of the LPID contain the frame sequence number. The frames are numbered consecutively, starting with 1 at the beginning of each course segment. The program uses the sequence number for reference when plotting the data points.

The program runs on a work tape, in regular history file sequence, containing only the student population wanted. The tape may be mounted on either tape drive.

1. The program reads a student record, using subroutine FAME, into the 134 character array IREC. If the end of file is encountered, the program checks for a new student number. If it is a new student, the program goes to 2 below. If it is not a new student, the program executes 3 below.

2. The program stores the student number, goes to a new page on the printer, and writes the page header, which includes the student number and the axis of the graph.

3. The program examines IREC(18) and IREC(19), which are the last two characters of the LPID. The numerical value of these two positions are found, K=(10 x IREC (18) + IREC (19), and the first character of the response LP is stored as a data point in LINE(K).

4. The array LINE is written, and the data point is then removed from the array. The program repeats 1 above.

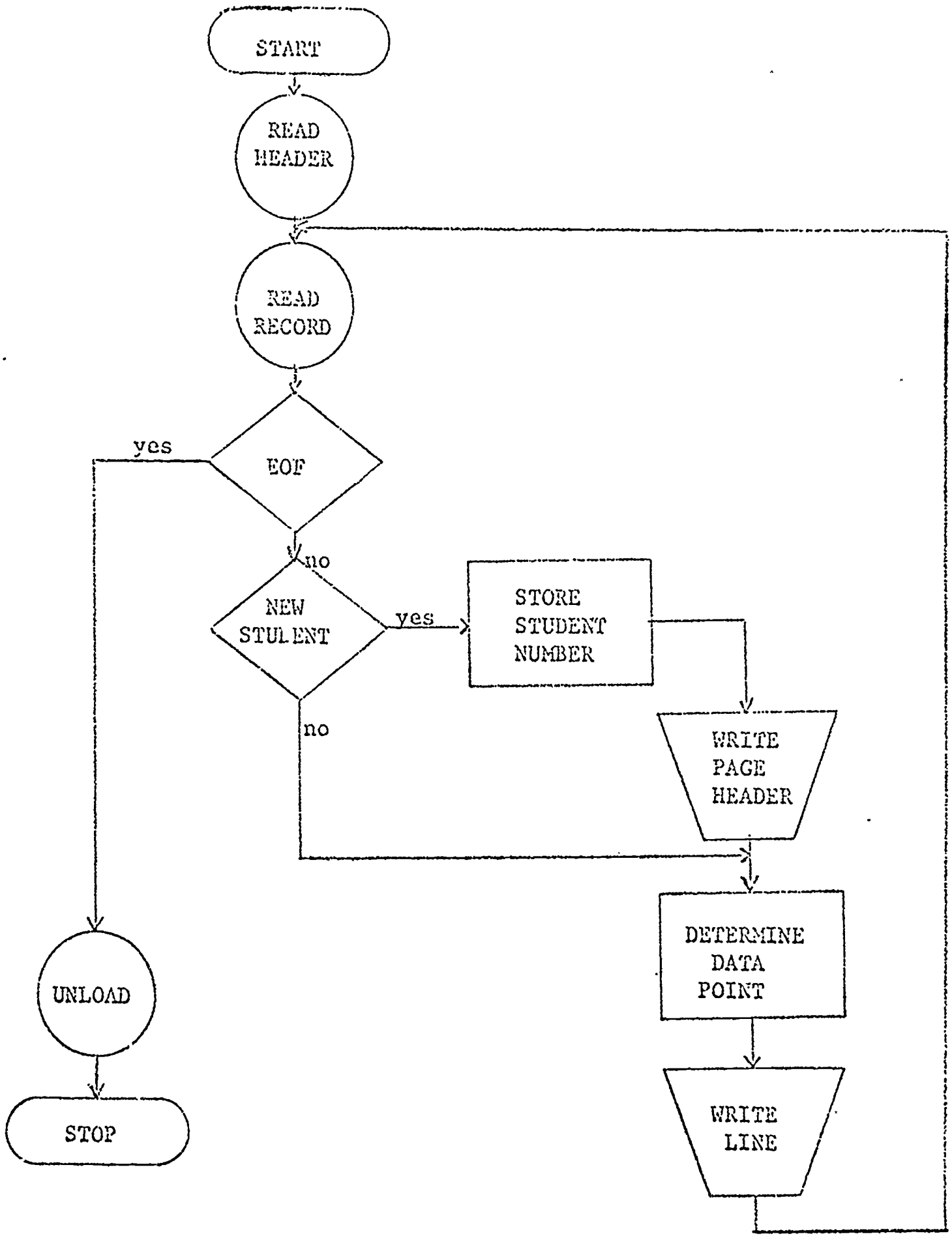


FIGURE 11
PROGRAM ISCST FLOW DIAGRAM

```
      DIMENSION NUM(10),LINE(102)                                       IST00060
C - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - IST00070
C                                                                       IST00080
C         THIS PROGRAM WAS WRITTEN BY                                   IST00090
C                   DAVID H. DASENBROCK                                 IST00100
C                   ISCS                                                IST00110
C                   OCTOBER  1968                                       IST00120
C                                                                       IST00130
C - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - IST00140
C       THE PURPOSE OF THIS PROGRAM IS TO TRACE A STUDENTS PATH THROUGH THE 00150
C  INSTRUCTIONAL MATERIALS.  THE X AXIS SERVES AS THE FRAME NUMBER, THE Y AS 160
C  THE RESPONSE NUMBER.  THE FIRST CHARACTER OF THE RESPONSE EP IS RECORED ON 70
C  THE PLOT.  A PLOT IS DRAWN FOR EACH STUDENT.  THE PROGRAM IS RUN ON A WORK 80
C  TAPE IN REGULAR HISTORY FILE SEQUENCE, SORTED BY STUDENT.             IST00190
C ANY EPID WITH 01 SEQUENCE NUMBERS APPEARING AS IN  IT SERVES AS A
C RESTART POINT. THIS INCLUDES EXCURSIONS, CHAPTER BEGINNINGS, OR ANY
C SEQUENCE WITH AN EPID OF THE FORM --------01.
C         LINE = ALPHANUMERIC ARRAY FOR LINE ON THE PRINTER, IN WHICH TO STORE 00
C       GRAPHIC CHARACTERS.                                             IST00210
C         NUM = ALPHANUMERIC ARRAY CONTAINING THE NUMBERS 0-9, FOR DETERMINING 20
C       THE X AXIS POSTION                                              IST00230
      COMMON IREC(134),JDATE(4)                                         IST00240
      DATA NUM/'0 ','1 ','2 ','3 ','4 ','5 ','6 ','7 ','8 ','9 '/      IST00250
      DATA IBLNK,IDOT,ID,N/' ','.','40',2/                              IST00260
C       SET THE TAPE DRIVE NUMBER                                       IST00270
      WRITE(2,3004)                                                     IST00280
3004  FORMAT(//,20X,'SET TAPE DRIVE SENSE SWITCH',/,                    IST00290
     *20X,' SS 4 ON = USE TAPE DRIVE 5',/,                              IST00300
     *20X,' SS 4 OFF= USE TAPE DRIVE 6',///////)                        IST00310
      PAUSE                                                             IST00320
C       JD IS USED AS A SWITCH TO SET THE DRIVE NUMBER                  IST00330
C       JD IS SET BY SENSE SWITCH FOUR                                  IST00340
      CALL SSWTCH(4,JD)                                                 IST00350
      GO TO (3001,3002),JD                                              IST00360
3001  IDRIV=5                                                           IST00370
      GO TO 19                                                          IST00380
3002  IDRIV=6                                                           IST00390
C       READ THE HEADER                                                 IST00400
19    CALL RHDR(IDRIV)                                                  IST00410
C       READ THE TAPE RECORD                                            IST00420
C FAKE IS THE  FORTRAN SUBROUTINE ON THE CAI SYSTEM THAT READS EACH
C STUDENT RECORD INTO A 134 CHARACTER ARRAY CALLED FAKE.
10    CALL FAKE(IDRIV,N)                                                IST00430
C       CHECK FOR THE END OF FILE                                       IST00440
      IF(N-2)5,11,5                                                     IST00450
C       CHECK FOR A NEW STUDENT                                         IST00460
11    IF(ID-IREC(5))12,40,12                                            IST00470
C       SET STUDENT NUMBER                                              IST00480
12    ID=IREC(5)                                                        IST00490
C       WRITE PAGE HEADER                                               IST00500
100     WRITE(2,101)                                                    IST00510
101   FORMAT(1H1)                                                       IST00520
      WRITE(2,120)IREC(1),IREC(2),IREC(3)                               IST00530
120   FORMAT(20X,3A2)                                                   IST00540
```

```
      WRITE(2,110)IREC(4),IREC(5) ,IREC(10)                              IST00550
110   FORMAT(35X,'STUDENT NUMBER  ',2A2,10X,'CHAPTER ',A2,/)             IST00560
      WRITE(2,106)                                                       IST00570
106   FORMAT(2X,'RESPONSE',40X,'FRAME')                                  IST00580
      WRITE(2,107)                                                       IST00590
107   FORMAT(1X,'NUMBER')                                                IST00600
C      WRITE THE GRAPHIC AXIS                                            IST00610
      WRITE(2,108)                                                       IST00620
      WRITE(2,105)                                                       IST00630
108   FORMAT(19X,'1111111111222222222233333333334444444444555555555566661ST00640
     *66666777777777888888888999999999')                                 IST00650
105   FORMAT(10X,'12345678901234567890123456789012345678901234567890123IST00660
     *5678901234567890123456789012345678901234567890123456789')          IST00670
C      FILL ARRAY LINE WITH DATA POINTS                                  IST00680
      DO 13 I=1,102                                                      IST00690
      LINE(I)=IDOT                                                       IST00700
13    CONTINUE                                                           IST00710
      J=0                                                                IST00720
      WRITE(2,102)J,LINE                                                 IST00730
102   FORMAT(2X,I5,2X,102A1)                                             IST00740
60    DO 14 I=2,101                                                      IST00750
      LINE(I)=IBLNK                                                      IST00760
14    CONTINUE                                                           IST00770
C      CHECK FOR THE FRAME NUMBER                                        IST00780
40    DO 45 I=1,10                                                       IST00790
      IF(IREC(18)-NUM(I))43,42,43                                        IST00800
42    IA=I-1                                                             IST00810
43    IF(IREC(19)-NUM(I)) 45,44,45                                       IST00820
44    IB=I-1                                                             IST00830
45    CONTINUE                                                           IST00840
      IF(IA-0)47,46,47                                                   IST00850
46    IF(IB-0)47,10,47                                                   IST00860
47    IC=IA*10+IB+1                                                      IST00870
      J=J+1                                                              IST00880
      LINE(IC)=IREC(20)                                                  IST00890
C      WRITE THE ALPHANUMERIC ARRAY, LINE                                IST00900
      WRITE(2,102)J,LINE                                                 IST00910
      LINE(IC)=IBLNK                                                     IST00920
      GO TO 10                                                           IST00930
C      END OF PROGRAM *  UNLOAD  *                                       IST00940
5     CALL UNLD(IDRIV)                                                   IST00950
      STOP                                                               IST00960
      END                                                                IST00970
// XEQ ISCST                                                             IST00980
```

PROGRAM ISCST

OPERATING INSTRUCTIONS

This program is designed to read a work tape, comprised of ISCS students in history file sequence, and provide a trace of the students progress through the instructional materials

To run the program

1. mount the work tape (either drive)
2. put stock paper in printer
3. load the fortran deck
4. set sense switch 4 to assign the proper tape drive (drive selection instructions appear on printer)

OUTPUT

1. tape drive selection instructions
2. trace of student path through the instructional materials.

PROGRAM ISCSB

PROGRAM LOGIC DESCRIPTION

This program was developed to fulfil specific data analysis needs of the Intermediate Science Curriculum Study. The ISCS-CAI program is completely macro based. As a result, the same response identifiers are used throughout the course to indicate answers to free response questions. The purpose of this program is to select free response questions from a work tape containing only the student population wanted, and sorted by an item analysis sequence. The program prints the student responses, by question.

1. The program reads the tape header.

2. The program reads one student record, using subroutine FAKE. The student record is stored in the 134 character array IREC. The program searches for a D in IREC(20) or an F in IREC(21). These two characters, D and F, are unique in those two positions for free response questions in the macro system.

3. If a D or F is not encountered, the program repeates 2 above.

4. If a D or F is encountered, the program goes to a new page on the printer, and writes the page header containing the LPID.

5. The LPID is stored.

6. The student number, response identifier, and the contents of buffer zero are printed.

7. The program reads the next student record. This LPID is compared with the stored LPID.

8. If it is the same, the program returns to 6 above.

9. If it is not the same, the program returns to 3 above.

After each read statement, an end of file check is made.

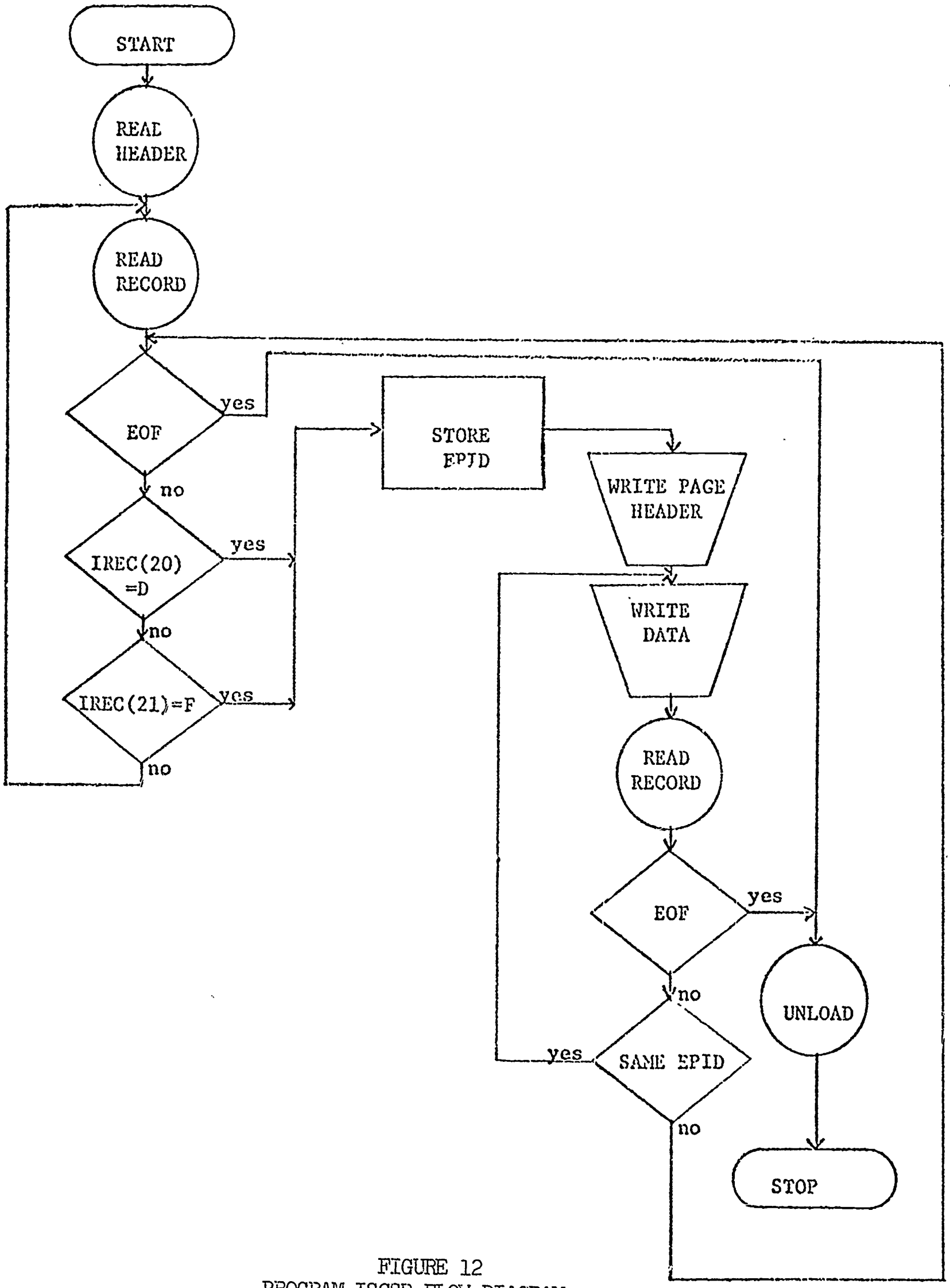


FIGURE 12
PROGRAM ISCSB FLOW DIAGRAM

```
      DIMENSION ID(10),IALP(26),IACPT(2)                                ISB00060
C*****************************************************************     ISB00070
C                                                                       ISB00080
C         THIS PROGRAM HAS WRITTEN BY                                   ISB00090
C                   DAVID H. DASENBROCK                                 ISB00100
C                   ISCS                                                ISB00110
C                   NOVEMBER  1968                                      ISB00120
C                                                                       ISB00130
C*****************************************************************     ISB00140
C                                                                       ISB00150
C     THE PURPOSE OF THIS PROGRAM IS TO SELECT FROM A WORK TAPE, SORTED INSB00160
C   ITEM ANANLYSIS SEQUENCE, THOSE QUESTIONS WHICH HAVE A FREE RESPONSE ANSWER.70
C   THE FREE RESPONSE EP IDENTIFIERS ARE DC, CF, WF, AND THE PROGRAM SEARCHES
C   FOR AN F AS THE SECOND CHARACTER IN THE EP RESPONSE IDENTIFIER, OR A D IN 190
C   THE FIRST CHARACTER OF THE IDENTIFIER, AS THESE ARE UNIQUE IN THE MACRO C0200
C   BEING USED BY ISCS.                                                 ISB00210
C         IREC = 134 CHARACTER ARRAY FOR THE TAPE RECORD                ISB00220
C   A STUDENT RECORD IS READ INTO THE IREC ARRAY USING SUBROUTINE FAKE.      221
C         ID = ARRAY FOR STORING AND CHECKING THE 10 CHARACTER EPID      ISB00230
C         IALP = ARRAY IN WHICH IS STORED THE 26 LETTERS OF THE ALPHABET,ISB00240
C       SETTING CHAPTERS IN THE PRINTOUT                                ISB00250
C         IACPT = ARRAY CONTAINING F AND D FOR SELECTION OF THE QUESTIONSISB00260
      COMMON IREC(134),JDATE(4)                                         ISB00270
C         DATA DECLARATIONS                                             ISB00280
1003  DATA IALP/'A ','B ','C ','D ','E ','F ','G ',  ,'H ','I ','J ', ISB00290
     *'K ','L ','M ','N ','O ','P ','Q ','R ','S ','T ','U ','V ','W ', ISB00300
     *'X ','Y ','Z '/                                                   ISB00310
      DATA IACPT/'F ','D '/                                             ISB00320
      DATA IR/'R '/                                                     ISB00330
C         INITIATE THE CORRECT TAPE DROVE.  JD IS USED A                ISB00340
C         INITIATE THE CORRECT TAPE DROVE.  JD IS USED AS A SWITCH TO SELISB00350
C       TAPE DRIVE, AND SET IDRIVE, THE TAPE DRIVE NUMBER  SENSE SWITCH 4 ISB00360
C       TO SET JD                                                       ISB00370
      WRITE(2,3004)                                                     ISB00380
3004  FORMAT(//,20X,'SET TAPE DRIVE SENSE SWITCH',/,                    ISB00390
     *20X,' SS 4 ON = USE TAPE DRIVE 5',/,                              ISB00400
     *20X,' SS 4 OFF= USE TAPE DRIVE 6',///////)                        ISB00410
      PAUSE                                                             ISB00420
      CALL SSWTCH(4,JD)                                                 ISB00430
      GO TO (3001,3002),JD                                              ISB00440
3001  IDRIV=5                                                           ISB00450
      GO TO 199                                                         ISB00460
3002  IDRIV=6                                                           ISB00470
C         READ THE HEADER                                               ISB00480
199   CALL RHDR(IDRIV)                                                  ISB00490
      WRITE(2,25)                                                       ISB00500
      N=2                                                               ISB00510
C         READ A TAPE RECORD                                            ISB00520
      CALL FAKE(IDRIV,N)                                                ISB00530
C         CHECK FOR END OF FILE OR TAPE ERROR                           ISB00540
      IF(N-2)23,10,23                                                   ISB00550
C         CHECK FOR CORRECT RESPONSE IDENTIFIER                         ISB00560
10    IF(IREC(21)-IACPT(1))19,20,19                                     IS
      IF(IREC(20)-IACPT(2))22,20,22                                     IS00580
```

```
C        STORE NEW EPID                                                          ISB00590
20     DO 11  I=1,10                                                             ISB00600
       ID(I)=IREC(I+9)                                                           ISB00610
11     CONTINUE                                                                  ISB00620
       DO 15 I=1,22                                                              ISB00630
       IF(IREC(10)-IALP(I))13,12,13                                              ISB00640
C        CHECK FOR CHAPTER                                                       ISB00650
12     IC=I                                                                      ISB00660
13     IF(IREC(11)-IALP(I))15,14,15                                              ISB00670
C        CHECK FOR PAGE                                                          ISB00680
14     IP=I                                                                      ISB00690
15     CONTINUE                                                                  ISB00700
C        CHECK FOR REVIEW                                                        ISB00710
       IF(IREC(20)-IR)30,22,30                                                   ISB00720
C        WRITE THE PAGE HEADER                                                   ISB00730
30     WRITE(2,16)IC,IP,IREC(16),IREC(17),ID                                     ISB00740
16     FORMAT(1H1,//,10X,'CHAPTER',I3,/,10X,'PAGE    ',I3,/,2X,'QUESTION',     ISB00750
      *2X,2A1,18X,'IDENTIFIER',/,35X,10A1)                                       ISB00760
17     DO 18 I=1,10                                                              ISB00770
C        CHECK FOR NEW EPID                                                      ISB00780
       IF(ID(I)-IREC(I+9))10,18,10                                               ISB00790
18     CONTINUE                                                                  ISB00800
C        WRITE THE RECORD, AS AT THIS POINT IT IS ACCEPTABLE                     ISB00810
       WRITE(2,21)IREC(4),IREC(5), IREC(20),IREC(21),(IREC(K),K=84,104)          ISB00820
21     FORMAT(10X,2A2,2X,2A1,2X,21A2)                                            ISB00830
C        READ A NEW RECORD                                                       ISB00840
22     CALL FAKE(5,N)                                                            ISB00850
       IF(N-2)23,17,23                                                           ISB00860
23     WRITE(2,25)                                                               ISB00870
25     FORMAT(1H1)                                                               ISB00880
C        END OF TAPE, -UNLOAD-                                                   ISB00890
1502   REWIND IDRIV                                                              ISB00900
       STOP                                                                      ISB00910
       END                                                                       ISB00920
// XEQ ISCSB                                                                    ISB00930
```

PROGRAM ISCSB

OPERATING INSTRUCTIONS

This program is designed to read an item analysis sorted work tape, comprised of ISCS students, and write student responses to free response questions.

To run the program.

1. mount the work tape (either drive)
2. put 1 ply stock paper in printer
3. load the fortran deck
4. set sense switch 4 to assign proper tape drive (drive selection instructions will be printed)

OUTPUT

1. Tape drive selection instructions.
2. Student number, response EP, buffer zero, of free response questions.

Program ISCSM or ISCS9

Note: Program ISCS9 is a modification of the series of programs labeled ISCSM, ISCS7, and ISCS8. The original (1967-68) version of these programs was labeled ISCS.)

Program Logic Description

It has been found that for revision purposes and other uses of the data from the CAI trial, a matrix is the most useful format in which to obtain data. Program ISCSM was developed to provide matrices of student response identifiers, latency times, numbers of reviews, and dates of response. The program prints each of the four types of data in a matrix ordered by student number and frame identifier (EPID).

1. The program reads the tape header.

2. One student record is read, using subroutine FAKE, and a check for end of file is made.

3. A check is made to determine whether a new EPID has been encountered.
   a. If a new EPID is encountered, the response for the frame is stored as the last response for that student and the latency is zeroed.
   b. If a new EPID is not encountered, a check is made for a new student number. If a new student number is not encountered, latency is added and a new record is read as in 2 above.
   c. If a new student number is encountered, the response is stored and the latency is zeroed.

4. A check is made to determine whether data for 8 frames have been stored.
   a. If data for 8 frames are stored, a response matrix is written and a check is made relative to the need to print or skip the latency, data and review matrices. The latter three matrices are skipped if data for questions only are included in the records. (This would normally occur if a sort for questions only had been made prior to the analysis.)
   b. A check is made for whether or not punched card output is desired, and if so, data in binary form are punched on cards.
   c. The EPID counter is reset to zero.

5. If data for less than 8 frames are stored, a check is made for whether the response is a review. If it is a review a new response record is read as in 2. If it is not a review, the response is stored as the first response for the student, the student number is stored, and then a new record is read.

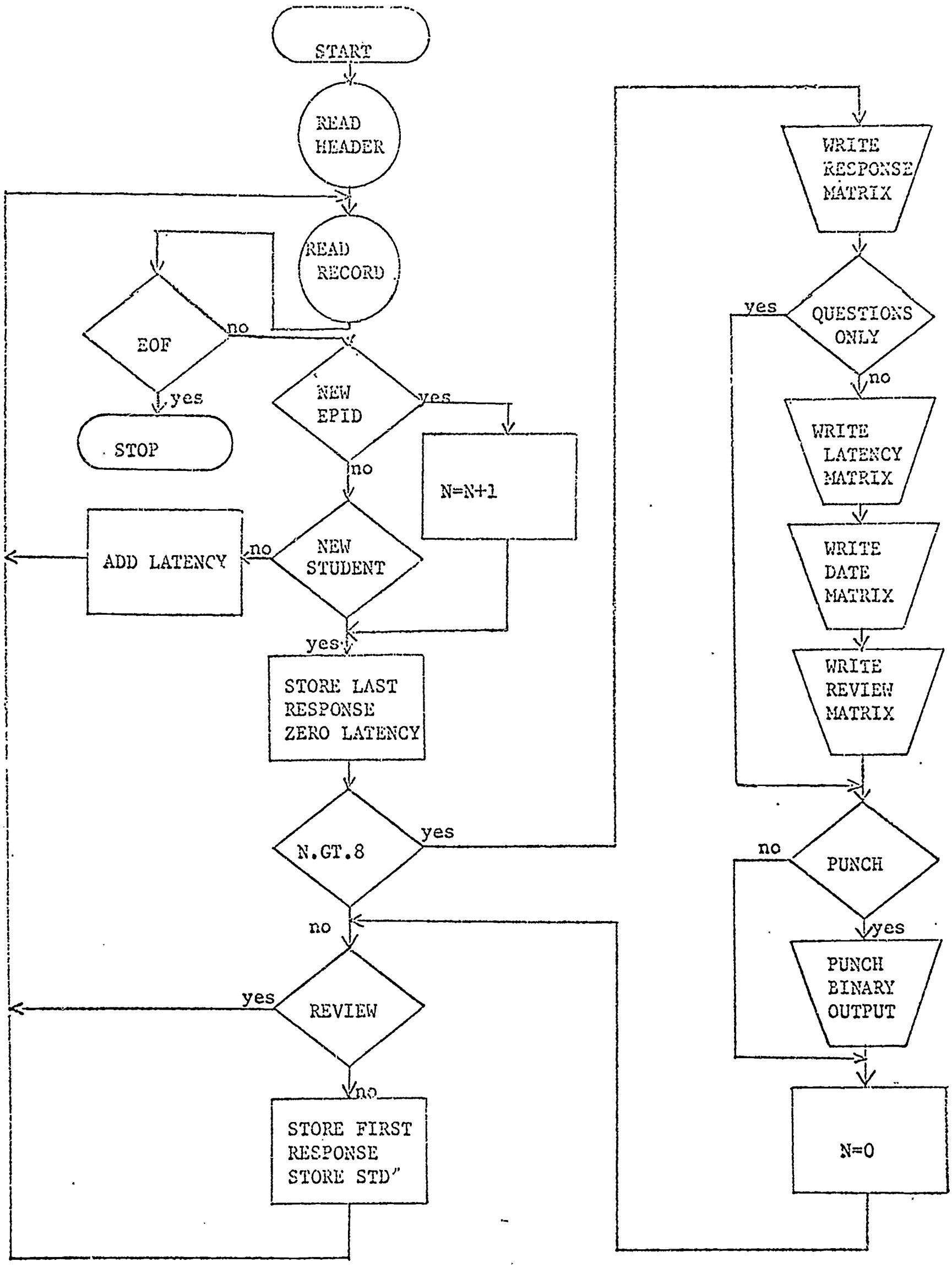


FIGURE 13
PROGRAM ISCS9 FLOW DIAGRAM

```
1001   INTEGER EPID(80),ANS(31,8,4)
1002   DIMENSION NUM(31),ICA(2),NLR(3),IQU(3),IBIN(62,8),LAT(31,8),
      *IDATE(31,8,4),IEPID(10),ICK(31,8),RAT(31,8) ,AVE(8)   ,
      *IREV(31,8),IREVT(8),NX1(16),X1(16)
1003   COMMON IREC(134),JDATE(4)
       DATA NS,NL,NC,NY/31,3,2,0/
C * * * * * * * * * * * * * * * * * * * * * * * * * * * *   *   * * * ** * * *
C
C         THIS PROGRAM WAS WRITTEN BY
C           DAVID DASENBROCK
C           ISCS
C           NOVEMBER, 1968
C
C           AND MODIFIED BY GEORGE HOGSHEAD
C           CAI CENTER
C           OCTOBER, 1969
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *  *
C
C   THIS PROGRAM IS DESIGNED TO PRESENT FOUR MATRICES OF DATA AS OUTPUT.
C  THE WORK TAPE FROM WHICH RECORDS ARE READ ARE SORTED IN ITEM ANALYSIS
C  SEQUENCE.  EACH STUDENT RECORD IS  READ AND THE AUTHENTICITY OF THE
C  STUDENT NUMBER IS CHECKED. TOTAL LATENCY AND AVERAGE LATENCY ARE
C  CALCULATED FOR EACH FRAME FOR EACH STUDENT AND STORED IN THE LATENCY
C  MATRIX.  ERROR RATES  ARE CALCULATED FOR EACH FRAME AND STORED IN THE
C  MATRIX WITH RESPONSE IDENTIFIERS. . FOUR MATRICES ARE PRINTED OUT...
C  RESPONSE , LATENCY, NUMBER OF REVIEWS, AND DATE OF RESPONSE....ALL IN
C  SEPARATE  MATRICES FOR WHICH  THE FRAME EPID APPEARS ALONG THE HORI-
C  ZONTAL AXIS AND STUDENT NUMBER  APPEARS ALONG THE VERTICAL AXIS.
C NS=NUMBER OF STUDENTS NA=BLANKS, NL= NUMBER OF ILEGAL RESPONSES, NC= NUMBER
C OF CORRECT ANSWERS, NY= END OF TAPE CHECK
C IBIN IS AN ARRAY OF BINARY NUMBERS FOR PUNCHING 1 AND 0 ON CARDS.
       DATA NUM/'00','01','02','03','04','05','06','07','08','09','10',
      *'11','12','13','14','15','16','17','18','19','20','21','22','23',
      *'24','25','26','27','28','29','30'/
C      NLR=OT,UN,RV
       DATA NLR/'U ','R ','O '/
C      ICA=CA,CF,EX
       DATA ICA,NA/'C ','E ','  '/
       DATA IQU/'X ','Y ','C '/
C THREE BLANK CARDS ARE READ TO CLEAR THE PROGRAM  FROM THE PUNCH AREA.
1005   READ(1,2001)IX
       READ(1,2001)IX
       READ(1,2001)IX
C NOTE TO OPERATOR TO SET SENSE SWITCHES.
       WRITE(2,2002)
       WRITE(2,2003)
       WRITE(2,2002)
1007   PAUSE
       CALL SSWTCH(2,JQ)
       CALL SSWTCH(3,JP)
C DETERMINE SENSE SWITCH SETTING FOR THE TAPE DRIVE IN USE. INSTRUCTIONS
C ARE PRINTED ON THE PRINTER.
       WRITE(2,3004)
```

```
3004  FORMAT(//,20X,'SET TAPE DRIVE SENSE SWITCH',/,
     *20X,' SS 4 ON = USE TAPE DRIVE 5',/,
     *20X,' SS 4 OFF= USE TAPE DRIVE 6',///////)
      PAUSE
C COMPUTER SETS DRIVE TO BE USED.
      CALL SSWTCH(4,JD)
      GO TO (3001,3002),JD
3001  IDRIV=5
      GO TO 3003
3002  IDRIV=6
3003  WRITE(2,2301)JQ,JP
2301    FORMAT(10X,2I4,///)
C READ TAPE HEADER.
      CALL RHDR(IDRIV)
      GO TO (10,15),JP
C WRITE STATEMENT TO TELL OPERATOR THAT THE PROGRAM WILL PUNCH CARDS.
10    WRITE(2,2004)
      PAUSE
C GO TO TOP OF PAGE.
15    WRITE(2,2005)
      LPCK=0
      LA=0
      M1=0
      N=2
C ZERO MATRICES.
19    DO 20 J=1,8
      DO 20 I=1,NS
      IREVT(J)=0
      AVE(J)=0
      IREV(I,J)=0
      LAT(I,J)=0
      ICK(I,J)=0
      DO 21 KA=1,4
C BLANK MATRICES.
      ANS(I,J,KA)=NA
      IDATE(I,J,KA)=NA
21    CONTINUE
20    CONTINUE
      DO173 I=1,16
      NX1(I)=0
173   CONTINUE
C SET BINARY MATRIX NUMBERS EQUAL TO 2, IF NO RESPONSE, 2 WILL REMAIN,
C IF THERE IS A RESPONSE, 1 OR 0 WILL BE PUNCHED.
      DO 110 J=1,8
      DO 110 I=1,62
      IBIN(I,J)=2
110   CONTINUE
      IPAGE=1
24    CALL FAKE(IDRIV,N)
CK EOF
      IF(N-2)1000,25,1000
CK NEW EPID
25    IF(M1-10)26,27,26
CK FOR QU
```

```
26      GO TO (50,51),JQ
50      DO 52 I=1,2
        IF(IREC(15)-IQU(I))52,51,52
52      CONTINUE
        GO TO 24
51      DO 29 I=1,10
        IEPID(I)=IREC(I+9)
29      CONTINUE
        M1=10
        NB=IREC(5)
        GO TO 28
27      DO 28 I=1,10
        IF(IEPID(I)-IREC(I+9))30,28,30
28      CONTINUE
CK NEW STD
C IREC%5  CONTAINS THE LAST TWO DIGITS OF A STUDENT NUMBER. NB CONTAINS
C A STORED STUDENT NUMBER. IF THEY ARE EQUAL, THE PROGRAM CONTINUES..
C IF THEY ARE NOT EQUAL, LATENCY IS RESET TO 0, AND THEN IT CONTINUES.
        IF(NB-IREC(5))101,31,101
101     LPCK=0
        LA =0
        ICK(NAME,IPAGE)=10
C LATENCY IS DIVIDED BY 60 TO GET IT INTO MINUTES. %THE COUNTER OVER-
C FLOWS IF LATENCY IS STORED IN SECONDS.  IT MAY OVERFLOW IN MINUTES,
C ON OCCASION, AND THE RESULT IS A NEGATIVE LATENCY.
31      LA=LA+IREC(22)/60
        IF(LPCK-10)103,152,103
152     ANS(NAME,IPAGE,3)=IREC(20)
        ANS(NAME,IPAGE,4)=IREC(21)
102     LAT(NAME,IPAGE)=LA
C THIS CHANGES LATENCY FROM AN INTEGER TO A REAL NUMBER.
        RAT(NAME,IPAGE)=LA*.1
        IF(NLR(2)-IREC(20))132,131,132
131     IREV(NAME,IPAGE)=IREV(NAME,IPAGE)+1
132     DO 106 I=1,NC
        IF(IREC(20)-ICA(I))106,107,106
107     IBIN(NAMB,IPAGE)=1
        GO TO 24
106     CONTINUE
        IBIN(NAMB,IPAGE)=0
        GO TO 24
103     NB=IREC(5)
        DO 33 I=1,NS
        NAME=I
        NAMB=NAME+31
        IF(NUM(I)-IREC(5))33,35,33
33      CONTINUE
C WRITES A STUDENT NUMBER WHICH DOES NOT MATCH A CORRECT ONE.
        LA=0
        WRITE(2,2006)IREC(4),IREC(5)
        GO TO 24
35      IF(ICK(NAME,IPAGE)-10)36,24,36
36      DO 32 I=1,NL
        IF(IREC(20)-NLR(I))32,102,32
```

```
32      CONTINUE
        ANS(NAME,IPAGE,1)=IREC(20)
        ANS(NAME,IPAGE,2)=IREC(21)
        DO 46 I=1,NC
        IF(IREC(20)-ICA(I))46,37,46
37      IBIN(NAME,IPAGE)=1
C FILL BINARY ARRAY.
        GO TO 38
46      CONTINUE
        IBIN(NAME,IPAGE)=0
38      CALL DATE
C DETERMINE DATE OF RESPONSE.
        DO 39 I=1,4
        IDATE(NAME,IPAGE,I)=JDATE(I)
39      CONTINUE
        LPCK=10
        GO TO 102
C FILL EPID IDENTIFIER ARRAY.
30      DO 41 I=1,10
        IB=(IPAGE*10)-10+I
        EPID(IB)=IEPID(I)
        IEPID(I)=IREC(I+9)
41      CONTINUE
        TOT=0.0
C TOTAL LATENCY ON THE FRAME.
        DO 127 I=1,31
        TOT=TOT+RAT(I,IPAGE)
127     CONTINUE
C DETERMINE THE AVERAGE LATENCY.
        AVE(IPAGE)=TOT/31.0
        LPCK=0
        LA=0
        M1=0
        KOUNT=IPAGE
        IPAGE=IPAGE+1
        IF(IPAGE-8)26,26,100
1000    NY=10
C PRINT RESPONSE MATRIX.
100     WRITE(2,2050)IREC(1),IREC(2),IREC(3)
        WRITE(2,2200)
        WRITE(2,2051)EPID
        DO 55 I=1,NS
        WRITE(2,2060)NUM(I),((ANS(I,J,K),K=1,4),J=1,KOUNT)
55      CONTINUE
        DO 171 I=1,NS
        DO 171 J=1,KOUNT
C DETERMINE ERROR RATE.
        L=J*2
        K=L-1
        M=I+31
        NX1(K)=NX1(K)+IBIN(I,J)
        NX1(L)=NX1(L)+IBIN(M,J)
        X1(K)=(31.0-NX1(K))/31.0
        X1(L)=(31.0-NX1(L))/31.0
```

```
      IF(X1(K)-.001)172,172,179
172   X1(K)=0.0
179   IF(X1(L)-.001)174,174,171
174   X1(L)=0.0
171   CONTINUE
      WRITE(2,2038)X1
2038  FORMAT(//,3X,'ERROR RATE',2X,8(1X,F4.2,1X,F4.2,2X))
      GO TO (86,87),JQ
C CHECK TO SEE IF OTHER MATRIX IS WANTED.
87    WRITE(2,2050)IREC(1),IREC(2),IREC(3)
C PRINT LATENCY MATRIX.
      WRITE(2,2201)
      WRITE(2,2051)EPID
      DO 60 I=1,NS
      WRITE(2,2061)NUM(I),(RAT(I,J),J=1,KOUNT)
60    CONTINUE
      WRITE(2,2020)AVE
2020  FORMAT(/,2X,'AVE LAT',8X,8(F6.2,6X))
C PRINT REVIEW MATRIX.
      DO 143  I=1,NS
      DO 143  J=1,KOUNT
      IREVT(J)=IREVT(J)+IREV(I,J)
143   CONTINUE
      WRITE(2,2050)IREC(1),IREC(2),IREC(3)
      WRITE(2,2033)
2033  FORMAT(40X,'NUMBER OF REVIEWS')
      WRITE(2,2051)EPID
      DO 141  I=1,NS
      WRITE(2,2030)NUM(I),(IREV(I,J),J=1,KOUNT)
141   CONTINUE
      WRITE(2,2031)(IREVT(I),I=1,KOUNT)
2030  FORMAT(6X,'VO',A2,7X,8(2X,I4,6X))
2031  FORMAT(//,6X,'TOTAL',6X,8(2X,I4,6X))
C PRINT DATE MATRIX.
      WRITE(2,2050)IREC(1),IREC(2),IREC(3)
      WRITE(2,2202)
      WRITE(2,2051)EPID
      DO 70 I=1,NS
      WRITE(2,2062)NUM(I),((IDATE(I,J,K),K=1,4),J=1,KOUNT)
70    CONTINUE
86    CONTINUE
81    WRITE(2,2005)
      GO TO (80,85),JP
80    DO 85 I=1,KOUNT
C PUNCH CARDS.
      IH=I*10
      NG=NS*2
      IW=IH-9
      WRITE(1,2100)(EPID(L),L=IW,IH),(IBIN(K,I),K=1,NG),IREC(3)
      GO TO (85,89),JQ
89    WRITE(1,2101)(EPID(L),L=IW,IH),(LAT(K,I),K=1,NS),IREC(3)
85    CONTINUE
      DO 88 I=1,80
      EPID(I)=NA
```

```
88     CONTINUE
1501   IF(NY-10)19,1500,19
1500   REWIND IDRIV
2001   FORMAT(A2)
2002   FORMAT(////)
2003   FORMAT(///,30X,'SET SENSE SWITCHES, SEE DATA REQUEST SHEET',///)
2004   FORMAT(////,10X,'THIS PROGRAM WILL PUNCH CARDS',//////////)
2005   FORMAT(1H1)
2006   FORMAT(/,10X,'STUDENT NUMBER  ',2A2,'  MISMATCH')
2050     FORMAT(1H1,///,50X,3A2)
2051   FORMAT(4X,'STUDENTS',4X,8(10A1,2X))
2060     FORMAT(6X,'VO',A2,6X,8(2X,2A1,2X,2A1,4X))
2061     FORMAT(6X,'VO',A2,7X,8(F6.1,6X))
2062   FORMAT(6X,'VO',A2,7X,8(4A2,4X))
2100     FORMAT(10A1,40I1,20X,A2)
2101     FORMAT(10A1,20I3,A2)
2200   FORMAT(//,45X,'MATCH IDENTIFIER')
2201   FORMAT(///,45X,'LATENCY IN MINUTES')
2202    FORMAT(///,45X,'DATE OF RESPONSE')
       STOP
       END
// XEQ ISCS9
```

PROGRAM ISCS 9*

OPERATING INSTRUCTIONS

The program is designed to print student response identifies latency, response date, and number of review in student by question matrices. The program will also punch the first and last pass responses in binary form. The work tape must be in item analysis sequences.

1. mount work tape (either drive)
2. put 1 ply stock paper in printer
3. load the fortran deck and 4 blank cards
4. set sense switch 4 to assign proper tape drive (drive selection instruction will be printed)
5. set sense switch 2 (on = questions only, off = all frames)
6. set sense switch 3 (on = punched output, off = no punched output)
7. load blank cards if punched output is desired

*Note: Program ISCS9 is a modification of the program ISCS7 and ISCS8. The original version of these three programs was labeled program ISCS.

APPENDIX

LAYOUT FOR ARRAY IREC